Matching the Knowledge Needs of Trade and Industry: Advanced and Operational Knowledge Based Systems

Larry Lucardie, Johan de Gelder & Cuno Duursma

ABSTRACT

Complex tasks that are being performed in trade and industry such as diagnosis, engineering and planning, increasingly require rapid and easy access to large amounts of complicated knowledge. To cope with these demands on trade and industry, advanced automated support for managing knowledge seems to be needed. Knowledge based systems are claimed to match these needs. However, to deal with the vast volume and complexity of knowledge through knowledge based systems, preconditions at three computer systems levels should be fulfilled.

At the first level, called the knowledge level, the development of knowledge based systems requires a well-elaborated theory of the nature of knowledge that helps to get a clear and consistent definition of knowledge. By providing guidelines for selecting and developing methodologies and for organising the mathematical functions underlying knowledge representation formalisms, such a definition significantly advances the process of knowledge engineering. Here, we present the theory of functional object-types as a theory of the nature of knowledge.

At the second level, called the symbol level, the representation formalisms used must be compatible with the chosen theory of the nature of knowledge. The representation formalisms also have to be interpretable as propositions representing knowledge, so that their knowledge level import can be assessed. Furthermore, knowledge representation formalisms have to play a causal role in the intelligent behaviour of the knowledge based system.

At the third level, called the systems level, a knowledge based system should be equipped with facilities that enable an effective management of the representation formalisms used. Yet other system facilities are needed to allow the knowledge base to communicate with existing computer systems used in the daily practice of trade and industry, for instance Database Management Systems, Geographical Information Systems and Computer Aided Design Systems. It should be taken into account that these systems may run in different networks and on different operating systems.

A real-world knowledge based system that operates in the field of soil contamination exemplifies the development of an advanced and operational knowledge-based system that complies with the preconditions at each computer systems level.

1. Knowledge And Knowledge Based Systems

Companies analyse their business process predominantly from an economic perspective. Economists, managers, and other policy makers tend to view the business process in terms of financial transactions. If the business performs badly they tend to propose actions that are in line with an economic view, e.g. cut backs in expenditure or cost cutting by optimisation or merging of divisions.

In recent years, the conviction is winning ground that, complementary to this economic perspective, a knowledge perspective should be adopted (Beerel, 1987; Lucardie, 1994; Steels, 1990). From a knowledge perspective one of the key assets held by any company is the human knowledge and reasoning ability: every worker is seen as a knowledge-worker and a staff member is valuable mainly because of his or her possession of a certain amount of knowledge. Through their knowledge, they can perform critical tasks that are important for the survival of a company. These include analysis, configuration, design, diagnosis, engineering, interpretation, monitoring and planning. From a knowledge perspective these crucial tasks are described and managed on the basis of the knowledge needed to perform them.

A phenomenon that interferes with an adequate fulfilment of tasks is knowledge erosion. Knowledge erosion is a collective name for all sorts of processes that lead to fragmentation of knowledge, unbalanced concentration and uncontrolled growth of knowledge. It leads to knowledge losses and ineffective uses of knowledge. As wind and water erosion can be devastating to rock and earth, knowledge erosion can be the cause of serious problems of companies.

The employment of knowledge based systems is claimed to have a strategic impact to cope with processes of knowledge erosion and to support knowledge management. Knowledge based systems distinguish themselves from other computer systems by an explicit representation of knowledge. Rather than concealing knowledge in program code, as is the case in many systems, knowledge is represented separately. Knowledge based systems are expected to provide leverage by performing knowledge management tasks such as:

- improving the understanding of knowledge
- providing easy access to knowledge
- offering facilities to maintain knowledge
- supporting decision making
- delivering facilities for creating, sharing and applying knowledge

Today, knowledge based systems have attained a permanent and secure role in industrial and commercial settings. In comparison to other technologies such as robotics or vision systems, knowledge based systems experience a rapid diffusion and uptake in world economy (Hayes-Roth & Jacobstein, 1994). However, to deal with the volume and complexity of knowledge through knowledge-based systems, conditions at three computer systems levels have to be fulfilled.

2. Computer Systems levels

The notion of computer systems levels occurs through computer science with varying degrees of utility and precision. Here, we will use a precise and useful distinction of three computer systems levels: the knowledge level, the symbol level and the system level. To be able to develop advanced and operational knowledge based systems matching the knowledge needs in trade and industry, a number of conditions should be fulfilled at each of these levels.

The Knowledge Level

In (Newell, 1981)1981 Newell proposed a new and distinct computer systems level which he called the knowledge level (Newell, 1981). The knowledge level includes semantics only and completely excludes user-interface and implementation aspects. The introduction of the knowledge level is primarily intended *to have a separate level to define the nature of knowledge.*

In many theories on the nature of knowledge *concepts* play an important role as classificatory and storage mechanisms. A concept has an *intension* and an *extension.* The intension of a concept is a set of constraints (or conditions) that should be satisfied by an object to belong to the class covered by the concept. The intension refers to an *object-type.* The extension of a concept consists of the set of objects complying with the object-type. The object-type dominates and determines the conceptualisation of an object.

Knowledge-based systems perform a *matching task*: they all apply constraints (the object-type) to a set of real-world referents (the objects) to obtain a match. An object matches if it can be classified as an object-type. For instance, the soil contamination system attempts to match 'contaminated site' (the object-type) with all sorts of sites (the objects). Every site that matches the conditions of 'contaminated site' can be classified as a contaminated site. As we can see from figure 1, the description of the object-type and the objects contains abstraction mechanisms such as generalisation, specialisation and aggregation.

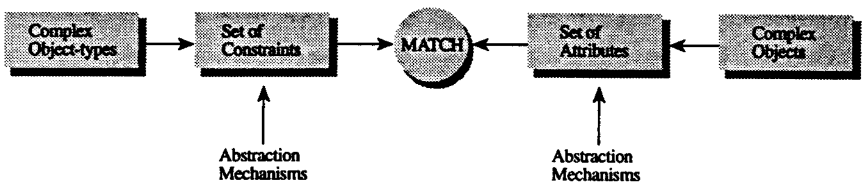
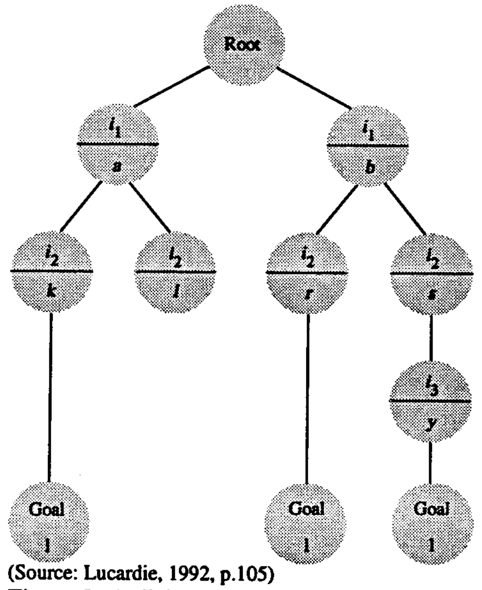


Figure 1: Matching Object-types and Objects.

There are several basic views on how to reconstruct the object-type of a concept, one of which is the functional view (Hendriks 1986; Lucardie, 1989; Reitsma; 1990; Van Der Smagt & Lucardie, 1991). According to the functional view an object-type of a concept is established by a goal-oriented reconstruction process in which a *disjunction of conjunct sets* is modelled. In the treelike schema of figure 2 such a disjunction is pictured. This disjunction consists of three conjunct sets all leading to goal 1:

{(i1;a),(i2;k)},
{(i1;b),(i2;r)} and
{(i1;b),(i2;s),(i3;y)}

An element of a conjunct set is an *inus-condition*: an Insufficient but Necessary part of the conjunct set which is Unnecessary but Sufficient for the result. Within a conjunct set an inus-condition is indispensable for achieving a goal, but the conjunct set itself, to which the inus-condition belongs, is replaceable by other conjunct sets.
As an object must satisfy one of the conjunct sets of an object-type in order to belong to the extension of a concept, the example, though greatly simplified, shows some interesting features of the functional approach. For instance, that it is possible for objects, which at first sight are different, to be identical in the context of a goal. Or, to put it in other words, objects having different attributes but matching with a conjunct set of an object-type, are equivalent. An example of two 'different' but functionally equivalent objects is formed by an object 1 characterised by the attributes {(i1;a),(i2;k)} and an object 2 having the attributes {(i1;b),(i2;s),(i3;y)}. In the context of goal 1 both objects are similar. Here, the notion of *functional equivalence* is essential: objects are identical, fall in the same concept or are similar if they possess -even quite different- attributes to perform the same function. Three mechanisms are responsible for the fact that a goal or a function is attainable by quite different strategies.



(Source: Lucardie, 1992, p.105)
Figure 2: A disjunction of three conjunct sets.

In the first place, the mechanism by which under certain conditions other attributes (descriptors) may become important for determining class membership. In the third conjunct set {(i3;y)} becomes a descriptor if {(i1;b),(i2:s)}. Figure 3 exemplifies conditional relevance in our soil contamination system.

Local emmission into dry soil by industrial section											
Local emission during production process	Low OR Medium OR High										None
Starting year	X < 1975					X >= 1975					-
Production process	-					Open				Closed	-
Production period	X < 10		X >= 10			X < 2		X >= 2		-	-
Toxicity of local emission of production process	High	Medium OR Low	High	Medium	Low	High	Medium OR Low	High OR Medium	Low	-	-
Local load on dry soil by industrial plant	Medium	Low	High	Medium	Low	Medium	Low	Medium	Low	Low	None

Figure 3: Example of conditional relevance in the soil contamination system: the attribute 'Toxicity of local emission of production process' has varying relevance. If 'Local emission during production process' is 'Low', 'Medium' or 'High' and 'Starting Year' equals or is greater than 1975 and 'Production process' is 'Closed', it is not relevant. The attribute 'Toxicity of local emission of production process' is not relevant in case 'Local emission during production process' is 'None'. Under all other circumstances, the attribute is relevant. (Remark: underlined attributes are detailed in another decision table).

In the second place, an important mechanism is that categorisations of attributes of objects influence each other. This phenomenon is called *conceptual interaction.* In figure 2 conceptual interaction manifests itself in the mutual influence of the categorisations of the first attribute and the second attribute. If {(i1;a)}, the classification of the second attribute is (K,L). If, on the other hand, {(i1;b)}, the classification of second attribute is (R,S) (see figure 4).

Local emmission into dry soil by industrial section											
Local emission during production process	Low OR Medium OR High										None
Starting year	X < 1975					X >= 1975					-
Production process	-					Open				Closed	-
Production period	X < 10		X >= 10			X < 2		X >= 2		-	-
Toxicity of local emission of production process	High	Medium OR Low	High	Medium	Low	High	Medium OR Low	High OR Medium	Low	-	-
Local load on dry soil by industrial plant	Medium	Low	High	Medium	Low	Medium	Low	Medium	Low	Low	None

Figure 4: Example of *conceptual interaction* in the soil contamination system: in one situation the attribute 'Production period' is classified as shorter than 10 years or longer than or equal to 10 years, in other situations 'Production period' is classified as shorter than 2 years or longer than or equal to 2 years. This varying categorisation interacts with 'Starting Year' and 'Production process'.

The third mechanism refers to the situation where objects may have different attribute values, but where this variation is limited to, or falls within, a goal-constructed category. Two objects characterised by the same {(i1;a)}, but with different values of {(i2;_)} -object 1 and object 2 respectively have the values k1 and k2 with k1 and k2 both falling in the category k of {(i2;_)}- are functionally equivalent (see figure 5).

<table>
<tr><td>Local emmission into dry soil by industrial section</td><td colspan="11"></td></tr>
<tr><td><u>Local emission during production process</u></td><td colspan="10">Low OR Medium OR High</td><td>None</td></tr>
<tr><td>Starting year</td><td colspan="5">X < 1975</td><td colspan="5">X >= 1975</td><td>-</td></tr>
<tr><td>Production process</td><td colspan="5">-</td><td colspan="4">Open</td><td>Closed</td><td>-</td></tr>
<tr><td>Production period</td><td colspan="2">X < 10</td><td colspan="3">X >= 10</td><td colspan="2">X < 2</td><td colspan="2">X >= 2</td><td>-</td><td>-</td></tr>
<tr><td><u>Toxicity of local emission of production process</u></td><td>High</td><td>Medium OR Low</td><td>High</td><td>Medium</td><td>Low</td><td>High</td><td>Medium OR Low</td><td>High OR Medium</td><td>Low</td><td>-</td><td>-</td></tr>
<tr><td>Local load on dry soil by industrial plant</td><td>Medium</td><td>Low</td><td>High</td><td>Medium</td><td>Low</td><td>Medium</td><td>Low</td><td>Medium</td><td>Low</td><td>Low</td><td>None</td></tr>
</table>

Figure 5: Example of *variation limited within goal-constructed categories* in the soil contamination system: An object with a low 'Local emission during production process' and a 'Starting Year' of 1979 and an open 'Production process' and a 'Production period' of 3 and a high toxicity is identical to an object with a low 'Local emission during production process' and a 'Starting Year' of 1990 and an open 'Production process' and a 'Production period' of 4 and a medium toxicity.

The Symbol Level

There is a close relationship between the knowledge level and the underlying symbol level. The central issue at the knowledge level is the nature of knowledge, whereas the symbol level is about its *representation* . Knowledge is reduced at the symbol level to structures and processes. The structures contain knowledge and the processes provide access to the captured knowledge. The following well-known equation summarises the idea of representation:

Knowledge Representation = Knowledge + Access
(structures) (processes)

Figure 6 below presents a few representation formalisms that originate from Artificial Intelligence and Database Technology:

Structure	Process
Logic	Theorem proving
Frames	Frame interpretation
Records	Query evaluation
Production rules	Backward/Forward chaining
Conceptual graphs	Restriction/joining/simplifying

Figure 6: A few representation formalisms from Artificial Intelligence and Database Technology.

The question remains how knowledge can be transferred into a representation formalism. Two strategies for solving this problem prevail. The first strategy is typified by a direct use of a representation formalism in the process of modelling knowledge. Herwijnen et al. (1990) is an exemplification of what we call the transfer approach. They advocate modelling knowledge by the direct application of record structures. In contrast, the second strategy employs intermediate techniques for the knowledge modelling process: Once the knowledge has been represented by the use of these techniques, the transfer to the target representation formalism is taken care of in a separate step. Davis (1988) gives an overview and analysis of a number of relevant techniques.

The transfer approach is not free from criticism. Many computer scientists and system developers are beginning to see that such computational answers are not completely satisfactory, because they often lead to low-level and short lived solutions. Knowledge representation and implementation issues are mixed which makes it difficult to see what exactly should be changed in these structures once the knowledge needs to be modified. Also Schreiber (1992) points to problems related to the transfer approach: difficult (or impossible) mapping of knowledge onto the required representation, difficult maintenance and poor explanation facilities. In the light of this criticism we may conclude that if the gap between knowledge and the target representation formalism is too wide, an intermediary and structured description of knowledge permitting us to conduct a knowledge level analysis is needed.

What conditions for the symbol level can be derived from the above? We think that the first condition is the use of a representation in which the theory used at the knowledge level can be effectively expressed. Having selected the theory of functional object-types as our approach, this implies that the techniques should not only offer expressive power to model functionally defined object-types and objects, but also provide associated validation facilities. The second precondition is that these techniques can easily be converted to the target representation formalism. The latter means that the target representation formalism is sufficiently powerful and flexible to contain the modelled knowledge. For an elaboration of constraints that in general should be met by a representation formalism see Smith (1985).

The Systems Level
At the systems level the representation formalism selected at the symbol level should be made available by an implemented management system Such a system should provide facilities centred around a specific representation formalism. DataBase Management Systems (DBMS's) and knowledge based systems shells are examples. DBMS's are usually record-based. They supply a host of facilities concentrated around records and query evaluation. These facilities may vary from basic data definition, data manipulation and associated administrative facilities to facilities for data security and integrity and database design and high-level tools for rapid application development. An example of a DBMS is ORACLE. Knowledge based systems shells are usually rule-based. Their facilities refer to rule-editors, inference engines for consulting the rules and tools to design a rule-based system. An example of an expert systems shell is OPS5.

A management systems does not only have to offer facilities for managing a specific representation formalism. They should also supply a variety of facilities to arrange communication with other systems that may be based on other representation formalisms. Section 3 describes an example of a database access extension built into the knowledge transfer system AKTS.

The above leads to the precondition that a management system should enable us to use a representation formalism including communication facilities.

3. Example: A KBS for the assessment of soil contamination

A century of industrial activities in Netherlands has left us with an inheritance of contaminated soil. Since 1980 the Dutch government has performed various tasks in support of the national soil clean up operation. A soil analysis is now required when sites are sold, new houses are built or inner cities are renovated. Research into historical industrial soil contamination is an essential part of this analysis: taking and testing of soil samples can be carried out more efficiently if one knows what contamination to expect.

The soil investigation process can be considered a knowledge intensive task: first of all the historical industrial activities at the site need to be listed. Once these have been identified the production process needs to be analysed to find the potential pollution hot spots. For instance, a car repair centre usually has a grease pit, a tank for used oil, and perhaps a paint spray workshop. Each of these processes may leave traces of certain chemicals. These chemicals may have contaminated the soil. The level of pollution depends on many factors including the type of production process, the type of factory floor, the toxicity of the chemicals, the reactions between the soil and the chemicals.

Knowledge analysis of soil contamination assessment.

The task of soil contamination assessment can be viewed as a matching process of objects (a specific site under investigation) to an object type (definition of a contaminated site). The soil experts know what object features need to be examined and how they should be used to classify a site (the object) as a contaminated site (the object-type). For instance, by combining the attributes of an object such as the period and duration of an industrial activity located at the site, the type of soil and its chemical behaviour, the experts can make their assessment of the contamination level of a site. The object-type 'contaminated site' is, of course, composed of a number of related, lower-level object-types. An example is the object-type 'Local load on dry soil by industrial plant'. Note that these object-types are defined according to the functional view. This implies a strong attention for functional equivalence as can be seen in the description of the object-type 'Local load on dry soil by industrial plant' (figures 3,4 and 5).

Thus the expert's analysis process can be viewed as a search through a structure of functionally defined object-types and matching relevant objects (sites) in order to assess the level of soil contamination. The knowledge level analysis resulted in a hierarchical system of functionally defined object-types.

Soil contamination assessment at the symbol level

At the symbol level the results of the knowledge level analysis are represented using a decision table system (figure 7). Figure 3, 4 and 5 show a decision table that is part of decision table system. This has been achieved through the 'Advanced Knowledge Transfer System' (abbreviated as AKTS; for an extensive description of AKTS see Lucardie, 1994). Using AKTS it was possible for the knowledge engineer to analyse and represent the expert's subtle considerations. Because of the fact that the decision table system contains no programming code, it is usable for discussing its contents with the experts. Figure 7 shows the hierarchical structure in which the tables are organised.

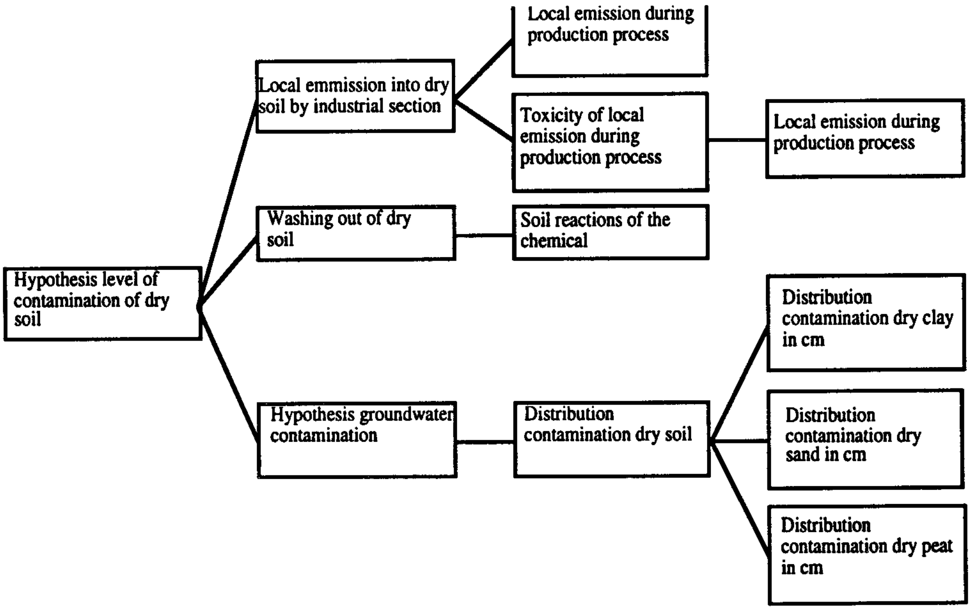


Figure 7: An overview of the decision table system that forms the knowledge base.

Apart from the decision table system an entity relationship diagram has been designed at the symbol level too. An investigation into the historical industries and their contamination in Brabant formed the starting point for this diagram. About seven entities have been included, among which industrial branch codes, chemical toxicity data, and soil reactions with the chemicals.

Soil contamination assessment at the systems level

The first version of the soil contamination assessment system is realised in the professional version of AKTS 1.0 for Macintosh. A database interface is added to AKTS for this project. Data from standard database packages can now be imported into AKTS and exported. A geographical index is included to allow the data to be displayed on a map. Initial experiments with MapInfo have been successful and appreciated by the users. After an intensive testing procedure the knowledge model will be used in a consult version of AKTS for Windows. The database will be integrated with the Soil Information System of the city of Rotterdam that runs under UNIX. Eventually the soil assessment system will be available as a Windows NT application with an on-line connection to this ORACLE database.

The maintenance of the knowledge stored in the decision table system will be perfomred by the soil experts, but will always be controlled by the knowledge engineers from TNO. Maintenance of the data in the database will be taken care of by the city of Rotterdam.

4. Epilogue

In this paper we argued that the development of a knowledge based system that can cope with the knowledge needs of trade and industry, should fulfill preconditions at three computer systems levels. At the knowledge level, we advocated the theory of functional object-types as a fruitful theory defining the nature of knowledge. At the symbol level, we used a decision table system, that is in line with our functional view, as the heart of knowledge representation. At the systems level, we employed AKTS as a management system revolving around the decision table system representation formalism. An overview of the development of a knowledge based system for soil contamination diagnosis has shown the effectiveness of our approach for supporting complex diagnosis tasks. In general, we expect that knowledge based systems, developed according to the described approach, will shape the pattern of intelligent computer applications in the near future.

Literature

Beerel, A. C. (1987). *Expert Systems. Strategic Implications and Applications.* New York: John Wiley & Sons.

Davis, A. M. (1988). A Comparison of Techniques for the Specification of External System Behavior. *Communications of the ACM, 31*(9), 1098-1115.

Hayes-Roth, F., & Jacobstein, N. (1994). The State of Knowledge-Based Systems. *Communications of the ACM, 37*(3), 27-39.

Lucardie, G. L. (1992). Functionele Object-typen als Grondslag voor Beslissingsondersteunende Systemen: De Technische Bouwregelgeving als Case. In H. De Swaan Arons, H. Koppelaar, & E. J. H. Kerckhoffs (Eds.), *NAIC'92*, (pp. 213-224). Delft: Delftse Universitaire Pers.

Lucardie, G. L. (1994). *Functional Object-types as a Foundation of Complex Knowledge-based Systems.* Computer Science, Technical University Eindhoven.

Newell, A. (1981). The Knowledge Level. *AI Magazine, 1*, 1-20.

Schreiber, A. T. (1992). *Pragmatics of the Knowledge Level.* Dissertation, Amsterdam, Faculty of Psychology.

Smith, B. C. (1985). Prologue to "Reflection and Semantics in a Procedural Language". In R. J. Brachman & H. J. Levesque (Eds.), *Readings in Knowledge Representation* (pp. 32-43). Los Altos: Morgan Kaufman.

Steels, L. (1990). Components of Expertise. *AI Magazine, 11*(2), 28-49.