FACILITATING THE CONSTRUCTION OF EXPERT SYSTEMS : METHODOLOGIES AND TOOLS

by

Asoka S. Karunananda

Abstract

Artificial Intelligence was born in 1956 with its main aim to develop intelligent artifacts. The construction of expert systems emerged as a main application area of Artificial Intelligence. The expert systems construction has still been an ad-hoc process rather than a discipline with a scientific basis. This paper discusses some methodologies and tools available for construction of expert systems. We also outline some of the current issues in the construction of expert systems and introduce the notion of Domain-Driven Knowledge Modelling as a way of addressing those issues.

1.0 Introduction

Expert system development has long been known as a main application area in Artificial Intelligence (AI). With its aim to developing the intelligent artifact, AI has to be aware of nature of human knowledge which has been a main theme of many other areas such as philosophy, psychology, cognitive science, epistemology, etc. Due to this mutidiciplinary influence on AI, the research in the construction of expert systems has also been affected drastically. This paper begins with a brief overview of what expert systems are. We proceeds to discuss methodologies and tools available to construct expert systems (more correctly Knowledge-Based Systems). We also point out some issues in KBS development which have not been fully addressed yet. Finally we briefly outline the notion of Domain- Driven Knowledge Modelling as a way of addressing some of the current issues in KBS development.

The rest of this paper is organised as follows. Section 2 provides a general overview of what an expert system is. Section 3 describes some historical/philosophical aspects of Knowledge Engineering and Knowledge Acquisition (KA) . This section also out line some issues in KA, Section 4 discusses some of the current KA methodologies. Section 5 discusses the current KA tools. Section 6 introduces the notion of domain-driven knowledge modelling as a means of addressing the issues mentioned in Section 3. Section 7 briefly describes our domain- driven knowledge modelling approach to utilizing metaphors. Section 8 summarises the paper.

2.0 What is an Expert System ?

Expert systems are information systems. They are used in a variety of disciplines such as medicine, engineering and scientific domains. The expert systems differ from classical database systems as the expert systems deal with knowledge rather than data. Data are very rigid and specific form of knowledge. What the experts use for solving problem is more than just facts and rules, but their experiences, commonsense, beliefs and even some form of incomplete knowledge. Knowledge is not only in multiple forms but also available from variety of sources. That means knowledge for constructing such systems is available from sources other than human experts. Based on this fact the terms Expert Systems was later changed as Knowledge-Based Systems (KBS). However, people interchangeably use these two terms.

Expert systems can operate on incomplete information and also uses heuristic knowledge. These systems can describe the level of certainty of the answer provided to a given question. They can also explain why and how an answer is made. Thus the expert systems are interactive.

The most important feature of the expert is known as the separation of knowledge base, the information to be used reach the solution, and the inference engine that manipulates the information in the knowledge base. This separation makes possible the incremental development of the system. That is the knowledge base can be updated independent of inference engine. Of course, this allows one to use same inference engine for constructing expert systems for different domain knowledge. This possibility yields the idea of *Expert*

Dr. A. S. Karunananda, BSc (Special), PhD, Senior Lecturer, Open University of Sri Lanka.

system shells. In this sense the main components of an expert system are known as knowledge base, *inference engine and user interface* through which a user can interact with it.

3.0 Knowledge Engineering and Knowledge Acquisition

In the early 1980s, the KBS development emerged as a distinct area of research. Feigenbaum (1980) named the new discipline as *Knowledge Engineering*. At the same time as the conception of Knowledge engineering area of Knowledge Acquisition emerged to facilitate the development of methodologies for constructing KBS. The following is a brief discussion of some philosophical aspects associated with the methodologies and the tools for KBS development.

3.1 Knowledge-level

As many researchers (e.g.Steels 1990, Chandrasekaran et al.1992) point out, in the early days, KBS development was mainly viewed in terms of programs or symbols, resulting in the view that an expert's knowledge usually takes the form of facts and rules. It has gradually become clear that research into expert system (or KBS) development should go beyond the level of formalisation and programming (Steels 1990). The key identification of such a level is explained in Newell's landmark article in the *Artificial Intelligence* journal (Newell 1982). In 1980, at the formation of the American Association of Artificial Intelligence (AAAI), in its first presidential address, Newell introduced the notion of the *knowledge level* as a new computer system level, see Figure 1.

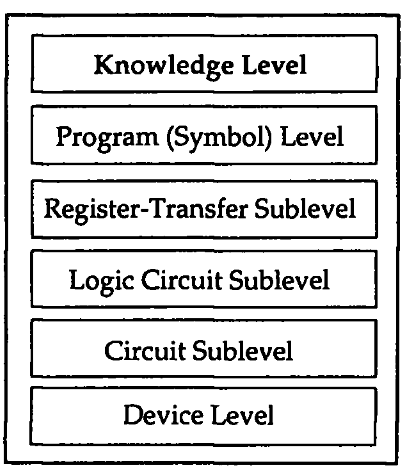

Figure 1 : Computer system levels (Newell 1982)

The *knowledge level* is more abstract than the program level. At this level, "knowledge is to be characterised entirely *functionally*, in terms of what it does, not structurally, in terms of physical objects with particular properties and relations" (Newell 1982,p.105). This hypothesis was a breakthrough in the area of knowledge acquisition.

3.2 Knowledge Modelling

The nature of the knowledge which is to be emulated on computers is central in KBS development. There are two main philosophical views of the nature of the expert's knowledge. These two views are known as *objectivism* and constructivism. The objectivist approach sees the expert's knowledge as a body of information which can be transferred, borrowed or stored (Nystedt 1983). In contrast, the constructivist view defines the knowledge as something to be modelled and it is not readily available. Unit recently, the objectivist view had won the argument (Chalmers, French & Hofstadter 1992,Clancey 1989). However, the KA community is now largely unanimous that this view is wrong and has taken the knowledge modelling perspective for KA. In line with this view the current KA methodologies/tools are known as knowledge modelling methodologies/tools rather than knowledge extraction methodologies/tools.

3.3 Some Essential Aspects of Knowledge Acquisition

The key activity of the construction of expert systems is seen as the acquisition of *knowledge acquisition*. The process is generally defined as an integration of several activities: elicitation of knowledge, interpretation of elicited knowledge and formalising the elicited data. Other researchers also have different views about the activities which include: the elicitation, analysis, synthesis, representation and implementation of knowledge. In general, elicitation, analysis, synthesis,are inter-twined activities (Morik 1989, Morik 1991, Nwana,Paton, Bench-Capon & Shave 1991, Nwana, Bench-Capon, Paton &Shave 1994). Knowledge representation has also been an important activity and research area in KA. The KA literature highlights two kinds of representations: *mediating representation and intermediate representation*. The term *mediating representation* is used to indicate "the sense of synthesis and coming to understand through the representation", while the term *intermediate representation* is defined as one "which only exists between flanking representations and is bound to them by clearly defined projection rules which map one interpretation to the next" (Johnson 1989,p.184). Ford, Canas & Jones (1991) and Nwana, Shave, Bench-Capon & Paton (1992) also have the same view. Some of the widely used knowledge representation methods include: *frames* (Minsky 1986), *semantic networks* (Quillian 1968), conceptual graphs (Sowa 1984), concept maps (Novak 1977) and SGN (Johnson 1989).

A detailed discussion on knowledge representations is beyond the scope of this paper. Our particular aim is to discuss KA methodologies and tools. However, all these methodologies and tools eventually need a type of representation (Newell 1982, Minsky 1986). The activity of implementation of knowledge is also strongly associated with the way we represent the knowledge.

3.4 Some Major Issues in Knowledge Acquisition

This section provides a critical review of some of the current issues in knowledge acquisition.

1. *Mental model harmonisation*: It is hard for the domain expert and the knowledge engineer to harmonise their mental models or come to share the same perspective of a domain (Recogzei & Plantinga 1987, Motta, Rajan & Eisenstadt 1990).
2. *Use of the proper tool at the proper time*: There often exists a mismatch between elicitation techniques and the structure of the problem domain, primarily as it is not usually clear from the literature what elicitation technique should be used, *where and when*.
3. *Making sense of acquired knowledge*: It is hard for a knowledge engineer to navigate and make sense of the large amount of information obtained during KA.
4. *Guidance during KA*: Little guidance is available to a knowledge engineer during knowledge acquisition. For example, "how is an application task classified, given some application task characteristics,and how are knowledge acquisition tools selected ?" (Nwana et al. 1994).
5. *A cognitive basis of knowledge modelling*: KA is unprincipled: there is as yet no theory of how to acquire human knowledge. The epistemological, cognitive and conceptual foundation of knowledge modelling leave much to be desired (Bradshaw & Woodward 1989).
6. *The need for a domain specification*: Due to the current unstructured approaches to knowledge acquisition, knowledge engineers move quickly to some cognitive definition of a domain and organisation without an adequate specification of a domain.

The issues outlined in this section are central to our research. All these issues (and more) in KA have been concisely classified in Shaw & Woodward (1990). According to their classification, KA issues mainly fall into 3 categories: philosophical issues, conceptual issues and methodological issues.

Philosophical issues in KA are seen as the contrast between objectivist and constructivist views of knowledge. Conceptual issues in KA are associated with conceptualisation and representation of knowledge of tasks and problem solving within the domain of interest (see also Clancey (1989). Researchers often refer to two types of models : mental models (Norman 1983) and conceptual models. Mental models are extremely abstract and therefore we usually deal with conceptual models. Shaw and Woodward state that "as the mental model is unavailable to an observer, the conceptual model is taken to represent the model used to address the task demand of the target system" (Shaw & Woodward 1990, p. 184) *Methodological issues* are associated with the confusion or mismatch between KA methodologies and their use for different types of knowledge involved. This is mainly because KA methodologies are dependent on the type of knowledge involved. Therefore, an investigation of the relationship between the tools and the type of knowledge in the domain of interest is required, before they are to be used.

The six problems discussed above can be placed into these three main categories (see Table 1).

Issues in Knowledge Acquisition		
Philosophical	Conceptual	Methodological
A cognitive basis of knowledge modelling	Mental model harmonisation	Use of the proper tool at the proper time
The need for a domain specification	Making sense of acquired know-ledge	Guidance during KA

Table 1: A Summary of Issues in KA

4.0 Knowledge Acquisition Methodologies

This section discusses some of the existing methodologies in knowledge acquisition and points out their contributions to addressing the issues outlined in the preceding section. KA literature (e.g Neale 1988, Karbach, Linster & Vo 1990, Karbach, Linster & Voss 1992, Fensel &van Harmelen 1994, Karunananda, Nwana & Brereton 1994) provides reviews of KA methodologies and tools on different bases. The KA methodologies are mainly twofold: first generation and the second generation methodologies.

The First generation methodologies are primarily defined as those which involve life cycle definitions for KBS development (Nwana et al. 1991). The main criticism of these methodologies is that none of them addresses knowledge acquisition as a separate stage in their life cycles. Prototyping was also commonly used in the development of first generation KBS. Normally prototyping is a technique for constructing a KBS but not a methodology (Nwana et al. 1994).

The Second generation methodologies emerged from the knowledge-level hypothesis. David, Krivine & Simmons (1993) introduced two special aspects of second generation expert systems or knowledge based systems. They include : modelling perspective with multiple models and they are based on the knowledge level hypothesis. Five main methodologies are now briefly discussed.

4.1 Inference Structures

As many researchers (e.g Steels 1990, Chandrasekaran et al.1992) point out, the inference structure was one of the earliest attempts to analyse knowledge: Clancey's (1985) heuristic classification and Chandrasekaran et al.'s (1992) hierarchical classification are well-known examples. The heuristic classification includes three major steps of inference: data abstraction, heuristic match and solution refinement, see Figure 2. Each of these stages involves a different type of knowledge. Almost all diagnostic expert systems use Clancey's heuristic classification methodology for reasoning.

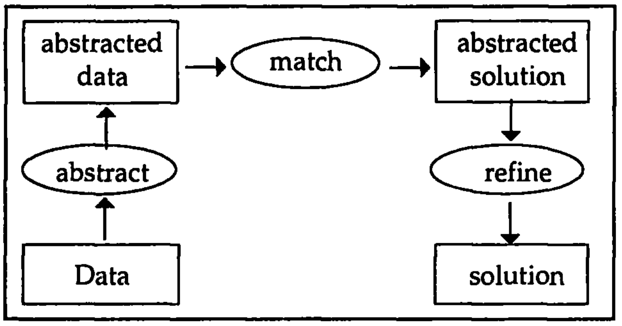

Figure 2: The Heuristic Classification Inference Structure (Clancey 1985)

4.2 Generic Tasks

The Generic Tasks (GT) approach assumes that the complex KBS reasoning task can be decomposed into a number of sub tasks (generic tasks). These generic tasks are associated with a type of knowledge and the family control regimes (Chandrasekaran 1985). Chandrasekaran argues that Generic tasks are the *building blocks* of intelligent systems (Chandarsekaran 1987, Bylander & Chandrasekaran 1988, Chandrasekaran & Johnson 1993). The term "generic task" is used to indicate the *problem solving types*. The following are some generic tasks (out of 6) identified Chandrasekaran (1985): *classification,State abstraction and refinement*. It is still debatable as to how far these tasks are indeed generic. As Chandrasekaran acknowledges, perhaps the most frequently asked question about GT is "How many generic tasks are there?" (Chandrasekaran & Johnson 1993). Chandrasekaran's answer to this question is that in the current perspective, there is no finite number of tasks or methods.

4.3 Role Limiting Methods

McDermott (1988) and associates proposed a methodology called *Role Limiting Methods* (RLM) for KBS development. These role-limiting methods emerged as a special case of *problem solving methods.* The literature often uses the terms problem-solving methods and role-*limiting methods* synonymously. This usage has been a convention and does not have any theoretical effect. However, the term role-limiting method has emerged in connection with the clarity of control knowledge attached to a method. McDermott (1988) defines the role-limiting methods as follows: "methods that strongly guide the knowledge collection and encoding will be called role-*limiting* methods".

McDermott (1988) and his co-workers tried to automate the acquisition of knowledge from experts. The main idea behind this approach is a set of *methods* for addressing different tasks. For example, MOLE (Eshelman, Ehret et al. 1987, McDermott 1988) is a tool that uses the *cover- and- differentiate* role-limiting method for the *diagnosis task*. The two major components of this method are : a set of problem states such as symptoms, complaints, etc. Which are to be covered and an exhaustive set of candidate explanations which *differentiate* among the elements in a set of problem states.

4.4 Components of Expertise

The knowledge analysis framework called *Component of Expertise* (COE) forms a synthesis of many of the existing methodologies (Steels 1990). This methodology integrates inference structures, problem solving methods, generic tasks, and deep versus surface knowledge. This methodology is applicable over a considerable range of domains since it integrates the features of several methodologies. This framework includes three main stages of knowledge analysis such as *task identification, investigation of possible methods and decomposition of tasks* (Steels 1990).

Steels (1990) also introduces the notion of *task structure* which shows the decomposition of tasks. This is slightly different from Chandrasekaran 's view of task structure is not just a decomposition of tasks, but also an integration of tasks, subtasks and methods, see Chandrasekaran & Johnson (1993).

4.5 The KADS Methodology: KADS -1

The KADS methodology (Wielinga, Schreiber & Breuker 1992), is considered as one of the most principled and comprehensive methodologies for knowledge acquisition. Originally, KADS was an acronym for "Knowledge Analysis and Documentation System". Recent interpretation of KADS is " Knowledge Analysis and Design Support" (Schreiber, Wielinga & Breuker 1993b). Todate the KADS methodology has gone through two major phases of research development. The first is now called KADS-1 (Wielinga et al. 1992), and its revision yielded the more recent KADS-2 (Schreiber et al. 1993a).

There are two central principles underlying the KADS methodology (Schreiber et al.1993 a):

- the introduction of multiple models as a means of reducing the complexity of the knowledge engineering process;
- the use of the knowledge-level descriptions.

The KADS methodology identifies seven distinct models : *the organisational model, the application model, the task model, the model of cooperation, the model of expertise, the conceptual model and the design model.* KADS incorporates the idea of knowledge level analysis through the introduction of the *model of expertise* (Schreiber et al. 1993a). KADS's epistemic view of expertise yields a four layer architecture for the model of *expertise.*

4.5.1 Four Layer Architecture of KADS Methodology

The four layer framework of the KADS methodology is discussed under the *model of expertise.* These four layers are *domain layer, inference layer, task layer and strategic layer,* Figure 3.

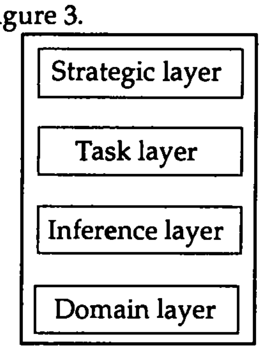

Figure 3: The Four Layer Architecture of the KADS Methodology

The domain layer consists of concepts, relations and structures of the domain (Breuker, Wielinga, van Someren, de Hoog, Schreiber, de Greef, Bredeweg, Wielemaker & Billault 1987). The description of knowledge at this layer is very similar to the static ontology in Alexander (1987). The Inference layer provides the functional description of problem solving. Information at the inference layer is organised into structures called *inference structures.* The inference structures in KADS consist of two primitives (Breuker et al.1987). They are the *knowledge sources* (functions) and the *metaclasses* (inputs and outputs of functions). For example, in Clancey's heuristic classification inference structure, Figure 2, *abstract* becomes a knowledge source in KADS while *data* and *abstracted data* are metaclasses for that particular knowledge source.

The inference structure in KADS is in fact the data-flow diagram specification used in conventional software design (Yourdon 1989). This is because knowledge sources play the role of data transforming operations while metaclasses play the role of data transforming operations while metaclasses play the role of data stores (Barbuceanu 1993). KADS attempts to define a more generic set of knowledge sources and metaclasses. Nevertheless, only a tentative classification of metaclasses and knowledge source can be found in Breuker et al.(1987).

The task layer of KADS deals with the ordering or organisation of inference structures for achieving particular goals. Such an ordering of inference structures is called a *task structure.* KADS's task structure includes three primitives: *goal, method and control knowledge* (Breuker et. al.1987).

The strategic layer is more abstract and consists of control knowledge that is required for tasks and inference layer activities. Many researchers believe that it is still possible to address many domains without considering this layer. (e.g. Breuker et al. 1988).

4.5.2 Interpretation Models

Beyond the four layer architecture, KADS has another unique feature over the other methodologies. It provides a library of *interpretation models* which can be instantiated for problem solving. The notion of interpretation models is somewhat similar to the idea of re-usability in traditional software development. The interpretation model library in the KADS methodology is still under development. KADS provides a library of interpretation models. It includes a taxonomy of threetype of real world problem solving task: *analysis, synthesis and modification.* Interpretation models for the analysis task have been considerably developed in KADS. However, interpretation models for the *synthesis* task have received much less attention. There are currently no interpretation models for the *modification* task.

4.6 The KADS Methodology: KADS-2

KADS is still under development and there has been some more recent improvements in the KADS methodology including the ongoing project called KADS-2. As many researchers point out, KADS-1 has been criticised for not providing proper support for descriptions at the domain layer (e.g. Bart Benus & van der Spek 1993, Schreiber 1993). This has led to a greater appreciation of domain specification for KBS development (Wielinga, de Velde, Schreiber & Akkermans 1993). In one attempt, KADS-2 has developed a *Domain Description Language* (DDL) that facilitates the description of the domain layer (Schreiber 1993). DDL extends the data modelling perspective of traditional software engineering; it is mainly an extension of Chen's (1976) Entity-Relationship modelling and KL-ONE (Brachman & Schmolze 1985).

The Common KADS modelling approach (Wielinga et al. 1993) defined in KADS-2 emerged as a synthesis of the traditional KADS four layer framework with the aspects of Components of Expertise approach (Steels 1990).

4.7 Discussion of KA Methodologies

Clearly, the approaches that involve a knowledge-level analysis give a better analytical description of knowledge leading to the development of KBSs. These methodologies are extremely powerful, provided that they are correctly used for the right domains. Due to the diversity of experts' knowledge in different domains, it is unreasonable to expect any methodology to be applicable to all domains. *The main limitation of the methodologies discussed thus far is that none of them provides a principled method to determine their applicability to a domain.* Such a decision is generally made by knowledge engineers, depending on their experiences and skills. This is a problem for the novice knowledge engineers. Further, none of these methodologies exactly *addresses* the issue of mental model harmonisation outlined in Section 3.4. It is quite obvious that KA methodologies must be more focussed on the human side of knowledge than they are at present. (Bell 1987, Bell 1988) Even KADS still does not provide the proper guidance to use its interpretation models so as to suit the nature of the domains.

Table 2 provides a summary of KA's key approaches.

5.0 Knowledge Acquisition Tools

This section discusses key knowledge acquisition tools cited in the literature. Here we discuss a selected set of tools under four categories: *specific systems related tools, repertory grids based tools, general tools* (or integrated systems) and machine learning tools.

5.1 Specific Systems Related Tools

Some KA tools are used to support the development, refinement and maintenance of specific KBSs. For instance, OPAL, (Musen, Fagan, Combs & Shortliffe 1987) supports the acquisition of knowledge on new cancer treatment plans for the expert system called ONCOCIN. TEIRESIAS (Davis 1976) was designed to develop and maintain the MYCIN expert system shell. In practicular, TEIRESIAS can find missing rules when the expert systems does not function as expected, Most of the automated KA tools discussed in McDermott

Feature	Methodology			
	KADS	CoE	GT	RLM
Underlying basis	Four layer architecture Interpretation Models	Integrated system Task Structure	Tasks can be decomposed Control knowledge	Tasks Methods
Task structure primitives	Goals Methods Control knowledge	Tasks Sub-tasks	Tasks Sub tasks Methods	Tasks Methods
Unique features	Interpretation models	Integrated system	PVCM to model task structure	Task specific

Tables 2: Summary of Key Methodologies

(1988) easily fall into this category as all of them are designed to perform some specific tasks utilizing the assigned methods. ROGET (Bennett 1985) is somewhat different: this is because ROGET deals with a generic model which can be applied to develop the expert systems for the domains which behave like the domain of MYCIN. The tools in this category are in fact considered as expert system shells. Note that the shells are not universally applicable for consructing KBS. This is because, a given shell is suitable for constructing a KBS only if the inference engine of the shell and the new domain have the same reasoning mechanisms.

5.2 Repertory Grids Based Tools

There are also some tools which are based on Kelly's Personal Construct Psychology and Ausubel's Assimilation Theory. In explaining the psychology of human thinking Kelly proposed the notion of construct in human communication (Kelly 1985). Constructs are bipolar entities (concepts) while concepts possess a single pole. Repertory grids-based KA tools are based on Kelly's Personal Construct Psychology. ETS and its successor AQUINAS Boose &bradshaw 1987) are well known examples. Hundreds of commercial expert systems are claimde to have been developed using ETS and AQUINAS. As Boose & Bradshaw (1987) state these systems include an AI tool advisor, a personal computer advisor and robotic tool selector. Many KA toolkits have a repertory grid as a main elicitation tool. The knowledge support system KSSO is an interactive KA systems which uses the reporty grid as a main tool (Shaw & Gaines 1987, Rappaport & Gaines 1990, Gaines & Shaw 1993).

ICONKAT is a tool which is based on both Kelly's Personal Construct theory and Ausubel's Assimilation cognitive learning theory (Ford et al. 1991, Ford Bradshaw, Admas-Webber & Agnew 1993). KRITON (Diederich, Ruhman & May 1987) also intergrates several of the elicitation tools including repertory grids. Nevertheless, KRITON has many similarties with the other KA tools which fall under the heading of general tools.

Our proposed approach to domain analysis has also evolved another novel way to use Kelly's (1955) personal construcut psychology for handling metaphors for knowledge modelling.

5.3 General Tools

There are some tools which are not designed for the purposes of a particular domain or a system. These tools generally integrate several tools in the form of a KA workbench. Some examples include KRITON (Diederich et al. 1987), keats -2 (Motta et al. 1990) and SHELLEY (Anjewierden & Wielemaker 1992). SHELLEY provides some supports for the KADS methodology; indeed SHELLEY is a CASE tool (Anjewierden et al. 1990). SHELLY bears many similarities to KEATS -2. Both emphasise the use of diffrent levels in analysis and provision of both top-down and bottom-up support in KA. The main diffference between them is that SHELLY's output is a documentation of the domain while KEATS -2 supports the development of the actual expert system.

KRITON is a fully automated KA workbench and it includes a set of knowledge elicitation tools. It automates the KA process by performing the elicitation, analysais and synthesis of knowledge. Like SHELLY and KEATS -2, KRITON also deals with different levels in KA process. However, KRITON has been criticised for lacking in theoretical justification (Motta et al.1990). There is also some research which aims to combine KRITON and KADS (Neale 1988). Steps have already been taken to combine Components of Expertise (implemented a CoE (Vawelkenhuysen & Rademakers 1992) and the KADS methodology under the KADS 2 - project (Wielinga et al. 1992) IKME provides another example of such an integtated system. (Barbucean 1993). It proposes a knowledge modelling language called MODEL. MODEL derives from the KADS methodology and the problem solving method *propose and revise* (McDermott1988).

5.4 Machine Learning Tools

A number of machine learning tools are also considered as types of KA tools. Chandrasekaran (1989) points out that learning is undoubtedly a means of acquiring knowledge. Tha main idea behind machine learning by identification of similarity. AQ11 (Michalski 1983) uses both positive and negative learning examples. Michalski points out that machine learning uses induction algorithms. It is clear that a significant knowledge engineering effort is needed to guide the induction algorithms. It is clear that a significant knowledge engineering effort is needeed to guide the induction algorithm to choose the right generalisation from a set of possibilities (Nwana et al. 1994). The main limitation of machine learning tools, as KA tools, is that their success is dependent on the availability of a large number of examples.

5.5 Discussion of KA Tools

We have superficially explored a wide range of KA tools. These tools are extremely powerful only when they are applied to the appropriate domains at the correct instant. The application of an unsuitable tool to a domain at the wrong time could lead to misinterpretation of the knowledge of the domain. It is important to research into the problem of selecting the best tool at the right time during the KA process. Figure 4 illustrates a selected set of tools which have a clear theoretical foundation.

Drawing from the discussion so far, it seems that the field of KA needs some pricipled guidance to make best use of the effective power of existing tools and methodologies. We have also understood that the applicability of tools/methodologies for KA is domain dependent activity. Therefore, we should analyse, organise or specify domains before using existing tools and methodologies. In other words, knowledge modelling should be driven by the nature of the domain.

6.0 Domain-Driven Knowledge Modelling

Thus far discussion reveals knowledge acquiition methodologies, tools and the evaluation of them. We also discussed the current issues which are not sufficiently adressed by these methodologies and tools. This section introduce the notion of Domain-Driven knowledge modelling a way of adressing these issues which were mentioned in Section 3.4.

Domain analysis is an aspect of knwledge-level analysis in KA. many research including conventional software engineers agree on the necessity for an analysis phase of a domain of interest before the design and implementation stages in their life cycles. Although there are some activities like requirements analysis (Yourdon 1989) and external analysis (KADS) in the literature, little attention has been paid to the development of domain analysis techniques. Nevertheless, extensive attention is paid to the other stages in the development life cycle. Here we use the term *domainanalysis* from a KA persepctive, which concerns the evolving nature of domains under the influence of theoretical, cognitive and epistemic considerations. This AI-based notion of domain analysis is slightly different from that used in software engineering or classical database development.

6.1 Review: Existing Domain Analysis Approaches

Using a domain analysis perspective to address the issues in Section 3.4 Woodward (1989) and Nwana et al. (1994) clearly delineate the phases of KBS development as shown in Figure 5

Tool Types	Underlying Methodology Theory / example Tools	KADS	Generic Tasks	Role Limiting Methods	Personal Construct Theory	Heuristics Classification	Hierarchical Classification	Use examples
General	Shelley	x						
	CoE		x	x				
	IGTT		x				x	
	CSRL		x				x	
	IKME	x			x			
	KRITON				x	x		
Repetory Grid Based	ETS				x			
	AQUINAS				x			
	KSSO				x			
	ICONKAT				x			
Specific	MOLE				x	x		
	MORE				x	x		
	SLAT				x	x		
Machine Learning	ID 3							x
	AQ11							x

Figure 4: Some tools and their associated theories

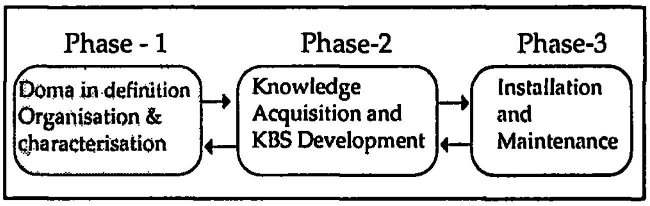

Figure 5: Phases in Knowledge Based System Development

Both (MEKAS) Nwana et al. (1994) and (COGNOSYS) Woodward (1990) frequently argue that the root cause of KA problems resides in the inadequate attention given to Phase-1 of Figure 5.

7.0 An Approach to Domain Analysis

A common feature of MEKAS and COGNOSYS is that both methodologies require the KE in the domain analysis process. As we postulate below, one rather ambitious aim in our domain analysis approach is to remove the knowledge engineer. Since we propose to capture the expert's perception of the domain, we believe that reducing / eliminating the knowledge engineer's role is important. *Otherwise we end up with KE's view of the expert's view of the domain.* We prefer to capture directly the expert's view without going via a knowledge engineer.

Another crucial contribution of our domain analysis approach is our exploitation of metaphors. MEKAS (Paton, Nwana, Shave & Bench - Capon 19921, Bench-Capon, Coenen, Nwana, Paton & Shave 1993) also propose exploiting metaphors for domain analysis, but they *Propose* no mechanisms for doing so. We argue that domain analysis is not just about determining the characteristics of a domain, but also involves capturing how experts see their domains. Research in psychology in the 1950s showed that *analogy making*, or the use of metaphors, is closely related to human perception. People often explain unknown things in terms of things already known meaning that there is a human tendency to anticipate the replication of event (Kelly 1955). The literature richly highlights the close connection between human perception and metaphors. For example, Minsky (1986) states that all our communication is metaphorical to some extent. MacCormac (1995) describes metaphors as cognitive devices. In their well-known book *Metaphors we live by*, Lakoff & Johnson (1980) state that metaphors are pervasive not only in language but also in thought and actions. Many other researchers have recently expressed their belief in the inseparable connection between analogy (a form of metaphor) and human perception (Carbonell 1982, Gentner 1985, Rumelhart & McClelland 1986, Chalmers et al. 1992). Feigenbaum, who introduced the term *knowledge engineering*, expressed the close connection between metaphors and the cognitive process as follows;

"Metaphorical reference appears to be omnipresent and almost continuous in our use of language. Further, if you believe that our use of language reflects our underlying cognitive process, then metaphor is a basis ideational process (Feigenbaum 1990, p.326)."

A most interesting feature of metaphors is that they lead two different people to interpret the same situation differently. Therefore metaphors can be used to reveal the nature of domain showing the experts own interpretation or perception of the domain.

The proposed approach to using metaphors for domain analysis is an integration of Kelly's (1955) *personal construct psychology* and Black' (1979) *interaction view of metaphors*. We introduce the notion of Personal Construct of Metaphors thus the approach is called PCM. Details of the approach can be found in Karunanda (1995) and Karunanda & Nwana (1955a).

We have implemented this approach using XPCE/ Prolog (Wielemaker & Anjewierden 1994). The implemented system is called DAKUM, an acronym for Domain Analysis for Knowledge acquisition Utilising Metaphors. It runs on sun Sparc Stations under Unix and X-windows. The system also runs on PCs running the Unix operating system with X-windows. An expert can directly interact with DAKUM without a knowledge engineer. While metaphor is the main modelling support in DAKUM, the system can also be used as on ordinary KA tool without using metaphors.

7.1 A Brief Illustration of the Dakum Domain Model

It would be very lengthy to fully illustrate how Dakum works. Rather, we provide a sample *domain model* from Dakum for the knowledge acquision domain. In this context, the term *domain model* is used to indicate the output generated by Dakum for a given domain. Karunananda (1995) fully describes how the domain model for knowledge acquisition domain was constructed using DAKUM. However, here we provide a brief description of the necessary aspects of a domain model generated by DAKUM.

Dakum is a software which allows the expert to directly interact with it and construct a domain model. In this process the tool captures the structural features and the functions of a domain. For example, if we consider the domain of personal computers, the structural features can be seen as monitor, disk drivers, keyboard, mouse, motherboard etc. Then functional description refers to the role of these structural components. Some function may be associated with more than one structural component. For instance, the function, *give inputs* is performed by the structural components, keyboard and the mouse. Nevertheless, keyboard may have a bigger role than mouse, as an input device. In this manner structural components can be associated with functions with different weights. DAKUM also facilitates the establishment of these associations between structural components and functions in a domain model. This aspect in DAKUM is called the mapping support. In this sense, DAKUM allows the expert to construct a domain model from a three dimensional perspective. The three dimensions refers to the structure, functions and the mapping between them.

Figure 6 illustrates a domain model generated by Dakum for describing the domain of *knowledge acquisition*. All children nodes of a particular parent form a block in 3D space, binding the structural features (S), functions (F) and their associated weights or ratings (R). Further elaboration of the nodes placed on the axis S, gradually yields the structural description. The functional axis F, refers to the functional description. The third dimension R, establishes the mapping between S and F.

It is quite easy to make sense of a domain model by referring to an illustration as depicted in Figure 6. For

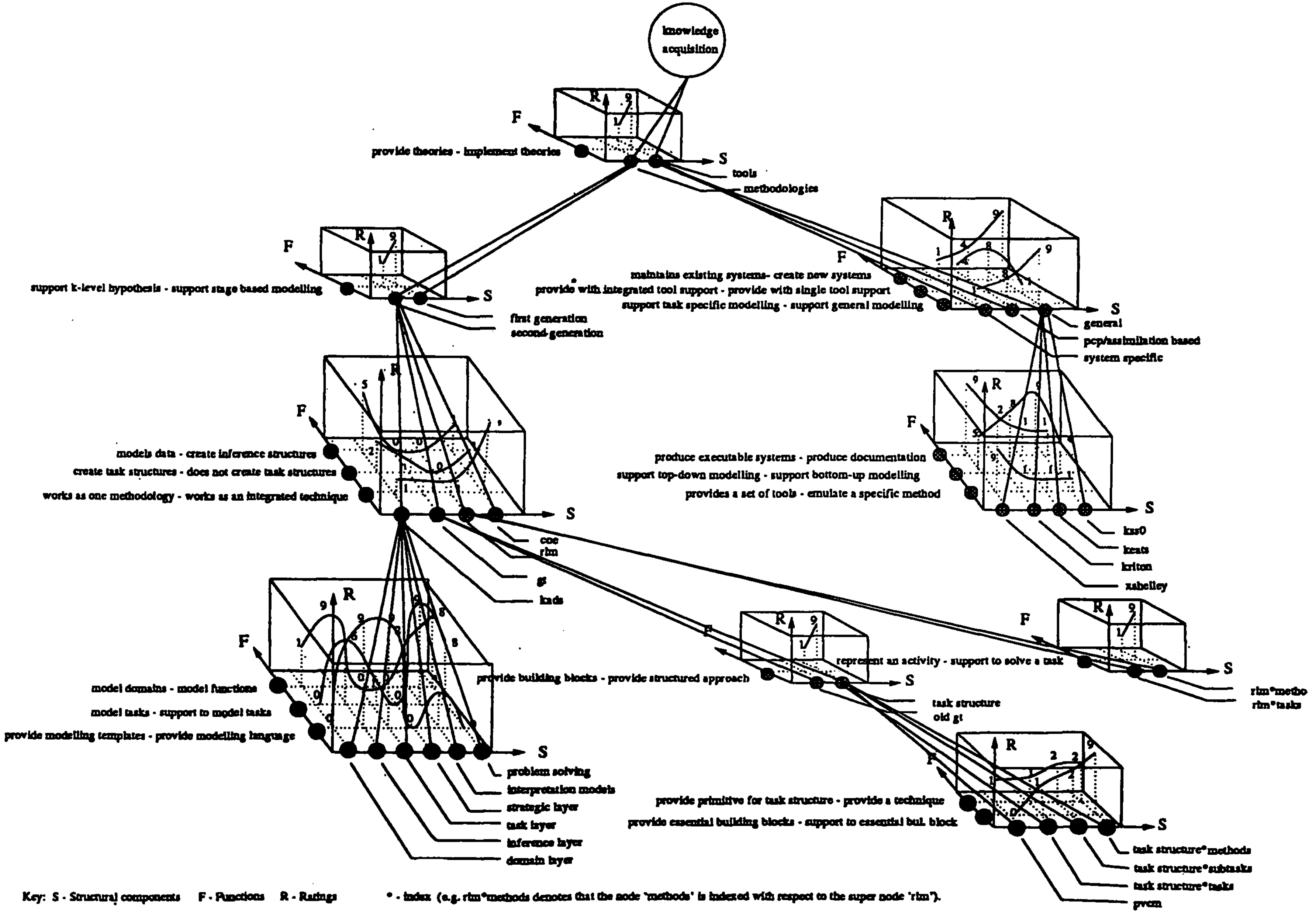

Figure 6
A domain model for K A domain

example, the top most 3D block says that knowledge acquisition can be structurally seen as having tools and *methodologies* (axis S). The functions of tools and methodologies are defined as provide theories-implement theories (axis F). Note that due to theoretical reasons, functions are indicated as dipole entities. There is also a scale defined on dipoles such that one (1) indicated the left most maximum weight while *nine* (9) is the right most maximum weight (axis R) According to this convention, methodologies perform the function, *provide theories* with weight 1, while tools perform the function, *implement theories* with weight 9 (See axis R). Other blocks should also be realised in the same manner.

Thus far, DAKUM has evolved through two major steps of preliminary evaluations and redevelopment. The first implementation (Karunananda & Nwana 1994), DAKUM version 1.1 was applied to one domain. It was illustrated in the papers presented in KAW' 95 (Karunananda, Nwana & Brereton 1995a), AISB'95 (Karunananda, Nwana & Brereton 1995b) and ECCS'95 (Karunananda & Nwana 1995b). An interested reader may refer to those papers for more details of our approach.

8.0 Summary and Further Work

This paper discussed knowledge acquisition in general with a particular emphasis on knowledge acquisition methodologies and tools. We pointed out some current issues in KA and critiqued the contribution of existing tools and methodologies to addressing these issues. In addressing the issues, we put forward a notion of domain analysis which captures an expert's perception of his/her domain. Finally we outlined some major aspects of our novel approach to Domain- Driven knowledge modeling.

We are in the process of continuing our research in Sri Lanka in the context of constructivism. In this project we intend to investigate the concept of *knowledge as a process* by means of a further study of a model of human behaviour which is presented in Karunananda (1991). We are also interested in facilitating the construction of expert systems for fuzzy domains. In this sense, we have initiated the construction of an expert system for diagnosis task in *Aurvedic Medicine*. This Project intends to enhance Calncey's heuristic classification idea, as Ayrvedic medicine is apparently very much heuristics based.

9.0 Acknowledgments:

This work was based on my research work in the Keele University, UK. I am grateful to Dr. H.S. Nwana for his guidance during my research work in Keele University UK. I am also grateful to Dr. O.P. Brereton (Keele, UK). I also acknowledge Prof. H. Sriyananda (OUSL) for his comments on the preparation of this paper for this journal. This work was supported by a grant from Association Commonwealth Universities (ACU).

10.0 Reference

Alexander, J.H. (1987) 'Ontological analysis: and ongoing *experiment' International Journal of* man Machine Studies 26, 473-485

Anjewierden A et. al. (1990), *Shelley User's Guide- version* 1.0 University of Amsterdam, Cap Sesa Innovation SD-Scicon.

Anjewierden A & Wielemaker, J (1992) 'Shelley-computer aided knowledge engineering' *Knowledge acquisition* 4, 109-125.

Barbuceanu, M (1993), 'Model: towards integrated knowledge modelling environments', *Knowldeg Acquisition* 5,245-304

Bart Benus, R d. H. & van der Spek, R. (1993) *The paint Advisor: A Communicative Expert System*, Academic press, Harcourt Brace Jovanovich Publishers, pp 267-288. In Guss Schreiber, Bob Wielinga & Joost Breuker (Eds) : Kads Principled Approach to Knowledge- Base System Development.

Bell J. (1987) 'The human side of knowledge engineering' *Educational Media International.*

Bell, J. (1988), The third role- *the naturalistic knowledge engineering*, Ellis Horwood Limited, Chichester, pp. 49-85 Dan Diaper (eds)

Bench- Capon, T., Coenen, F., Nwana, H Paton R. & Shave, M. (1993), 'Two aspects of the validation and verification of knowledge- based systems' *IEEE EXPERT* pp 76- 80.

Bennett, J.S. (1985) 'Roget a knowledge - based system for acquiring the conceptual structure of diagnostic expert system,', *Journal of Automated Reasoning* 1, 49-74.

Black, M. (1979) *More About Metaphor*, Cambridge University Press, Cambridge, London, pp. 19-43 In Andrew Ortony (Eds): Metaphor and Thought.

Boose, J.H. & Bradshaw,J.M. (1987) 'Expertise transfer and and complex problems: using AQUINAS as a knowledge acquisition workbench for knowledge- based systems.', *International Journal of Man Machine Studies* 26,3-28

Brachman, R.& Schmolze J. (1985) ' An overview of the KL-ONE knowledge representation system', *Cognitive science* 9, 171-216

Bradshaw, J& Woodward, N. (1989) *knowledge Acquisition Tools (Panel Discussion Summary of IJCAI 89 Knowledge Acquisition Workshop)* In Boose, J and Gaines, B (Eds,) Knowledge Acquisition for KBS Newsletter, 1(3) 12

Breuker J . et al. (1988) 'modelling in KBS development, A paper presented at the second European Knowledge Acquisition workshop EKAW' 88, Bonn

Breuker, J. Wielemaker, J. & Billault J-P. (1987), Model- driven knowledge acquisition: Interpretation models., Technical report, Social Science informatics, University of Amsterdam. Technical report Esprit Project 1098.

Bylander, T.& Chandrasekaran, B (19988), *Generic tasks for knowledge- based reasoning 'the right level of abstraction for knowledge acquisition'* Vol. 1, pp. 65-77. In B.R. Gaines & J.H. Boose (reds) knowledge Acquisition for Knowledge- Based Systems.

Carbonell J. G. (1982) *Metaphor: an inescapable phenomena in natural language comprehension*, Lawrence Erlbaum Associated, hillsdale, New Jersey, London pp. 415-434 in Lehnert and Right (eds.

Chalmers, D.J., French, R. M. & Hofstadter, D.R. (1992), 'High-level perception representation and analogy critiques of artificial intelligence methodology' *JETAI* 4 (3), 185-211.

Chandrasekaran, B. (1985) 'Generic task in knowledge- based reasoning Characterising and designing expert systems at 'right ' level of abstraction' Ieee pp. 294-300.

Chandrasekarn, B (1987), Towards a functional architecture for intelligence based on generic information processing tasks, *in 'the* proceedings of the Tenth International Joint Conference on Artificial intelli genece, (ed. McDermott)' Morgan Kaufman Publishers Inc, Los Altos, California, pp. 1183-1192.

Chandrasekaran, B. (1989) ' Task- structures, knowledge acquisition and machine learning' *Machine Learning* 4, 33-345.

Chandrasekaran, B. et al. (1992) 'Task- - structure analysis for knowldeg modelling', *Communication of the ACM* 35 (9) 124-136.

Chandrasekaran, B & Johnson, T R. (1993), *Generic Tasks and Task Structures: History Critique and New Directions*, springer- Verlag, Berlin, pp. 233-272 In: J. M. David and J. P. Krivine and R Simmons (Eds): Second Generation Expert Systems.

Chen, p.P.-s (1076) The entity- relationship model-towards a unification view of data' ACM *Transactions on* Database Systems 1 (1), 9-36.

Clancey, W.J. (1985) 'Heuristic classification' *Artificial Intelligence* 27, 289-350.

Clancey, W.J. (1989) ' The knowledge level reinterpreted: Modelling how system interact', *Machine Learning* 4, 285-291.

David, J.M., Krivine, J.P. & Simmons, R (1993), *Second Generation Expert Systems: A Step Forward in Knowledge Engineering, Springer* - Verlag, Berlin, pp. 3-23. In: M. David and J.P. Krivine and R. Simmons (Eds.): Second Generation Expert Systems.

Davis, R. (1976), Applications of Metal level Knowledge to the Construction. maintenance and Use of large knowledge Bases, Ph.d. thesis, technical report stan - cs-76-564, Stanford University, Stanford, CA.

Diederich, J., Ruhman, I.& May, M. (1987), 'KRITON: a knowledge acquisition tool for expert systems' *International Journal of Man Machine Studies* 26, 29-40.

Eshelman, L., Ehret, D.et. al. (1987), 'MOLE: a tenacious knowledge acquisition tool', International Journal of man Machine Studies 26, 41-54.

Feigenbaum, E. (1980), Knowledge engineering: the applied side of artificial intelligence, Technical report, Department of Computer Science, Stanford University . STAN- CS-80-812.

Feigenbaum, E.A. (1990), *Knowledge Processing: From File Services to Knowledge Servers*, MIT Press Cambridge, Massachusetts, London, England., pp. 324-329. In Raymond Kurzweil: The Age of Intelligent Machines.

Fensel, D. & van Harmelen, F. (1994), 'A comparison of languages which operatonalise and formalise KADS models of expertise', *The Knowledge Engineering Review*. Also: AIFB Techreport no. 280. University of Karlsruhe (Sep '93).

Ford, K.M., Bradshaw, J.M., Adams - Webber, J.R. & Agnew, N.M. (1993), 'Knowledge acquisition as a constructive modelling activity', *Inetrnational Journal of Intelligent Systems* 8 (1), 9-32.

Ford, K.M., Canas, A. & Jones, J. (1991), 'ICONKAT: an integrated constructivist knowledge acquisition *tool*', *Knowledge Acquisition* 3, 215-236.

Gaines, B.R. & Shaw, M.L.G. (1993), ' Eliciting Knowledge and transferring it effectively to a knowledge based system', IEEE *Transactions on Knowledge and data Engineering* 5(1), 4-14.

Gentner, D. (1985), 'Structure - mapping: A theoretical framework for analogy', *Cognitive Science* 7, 155-170.

Johnson, N.E. (1989), *Mediating Representation in Knowledge Elicitation* Chichester: Ellis Horwood chapetr 4, pp.179-194. In Diaper D. (ed), Knowledge Elicitation: Principles, Techniques and Applications Chichester: Ellis Horwood, 179-194.

Karbach, W., Linster, M. & Vo, A. (1990), *Modles of Problems - Solving: One label - One iden* IOS Press, Amsterdam, pp. 173-189. In Bob Wielinga, John Boose, Brian Gaines, Guus Schreiber and Maarten van Somaran (Eds): Current Trends in Knowledge Acquisition.

Karbach, W., Linster, M. & Voss, A. (1992), 'Models, methods, roles and tasks: many labels - one idea?', *Knowledge Acquisition* 2, 279-299.

Karunanada, A.S. (1991), Computer modelling of the theravadic Buddhist theory of thought processes a model of human behavior, Master's thesis, the Open University of Sri Lanka, Nawala, Nugegoda, Sri Lanka.

Karunananda, A. S. (1995), Investigating the use of metaphors for knowledge acquisition, Ph.D. thesis University of Keele, England.

Karunananda, A.S. & Nwana, H.S.(1994), A cognitive - based approach to using metaphors in domain analysis for knowledge acquisition, Technical Report TR 94-14, Department of Computer Science, University of Keele, Keele.

Karunananda, A.S. & Nwana. H.S. (1995a), 'An approach to using metaphors for knowledge acquisition'. A paper submitted to IJHCS.

Karunananda, A.S. & Nwana, H.S. (1995b), Cognitive - based approach to using metaphors in domain analysis for knowledge acquisition, in 'Proceedings of ECCS - 95', Saint - Malo, France.

Karunananda, A.S. Nwana, H.S. & Bereton, O.P. (1994), Towards a domain analysis approach utilising metaphors for knowledge acquisition, Technical Report TR 94-10, Department of Computer Science, University of Keele, Keele.

Karunananda, A.S., Nwana, H.S. & Brereton, p. (1995b), *Knowledge Acquisition Using Metaphors*, IOS Press, pp. 121-132. Hybrid Problems, Hybrid Solution, Edited by J. Hallam.

Kelly, G. A. (1995), *The Psychology of Personal Constructs*, New York, Norton.

Lakoff, G. & Johnson, M. (1980), *Metaphors we live by*, Chicago: University Press.

MacCormac, E.R. (1985), *Acognitive Theory of Metaphor*, MIT Press, Cambridge, MA.

McDermott, J.(1988) *Preliminary steps towards a taxonomy of problem Solving methods.*, Boston, Kluwer S. Marcus. Ed. : Automating Knowledge Acquisition for Expert Systems.

Michalski, R.S. (1993). *Theory and Method of Inductive learning*, New York: Tioga. In Michalski R. S. Carbonell J.G. and Mitchell T.M. (Eds) , Machine Learning: AN Artificial Intelligence approach.

Minsky, M. (1986), *Society of Mind*, *New* York, Simon and Schuster.

Morik. K. (1989), *Sloppy Modelling*, Springer Verlag, pp. 107-134. In Morik, K. (Eds), Knowledge Representation and Organisation in Machine Learning: Lecture Notes in Artificial Intelligence, 347, London.

Morik, K. (1991), 'Underlying assumption of knowledge acquisition and machine learning,' *Knowledge Acquisition* 3, 137-156.

Motta, E. Rajan, T. & Eisenstadt, M. (1990), 'Knowledge acquisition as a process of model refinement', *Knowledge Acquisition* 2, 21-49.

Musen, M., Fagan, L. M., Combs, D.M. & Shortliffe, E.H. (1987), 'Use of domain model to drive an interactive knowledge - editing tool', *International Journal Man - Machine Studies* 26. 105-121.

Neale, I.M. (1988), 'First generation expert systems: A review of knowledge acquisition methodologies', *know. Engi.* Rev. 3, 105-145.

Newell, A. (1982), ' The Knowledge level', *Artificial Intelligence* 18, 87-127.

Norman, D. A. (1983), *Some observations on Mental Models, Hillsdale,* NJ: Erlbaum.

Genter D. and Stevens A. Ed. : Mental Models.

Novak, J.D. (1977), A *Theory of Education*, Ithaca, NY: Comell University Press.

Nwana, H.s. Beach-Capon T.J. M. Paton, R.C. & Shave, M.J. R. (1994) 'Domain-dirven Knowledge modelling for knowledge acquisiton', *Knowledge Acquisition* 6, 243-270.

Nwana, H.S., Paton, R.C., Bench - Capon, T.J.M. & Shave, M.J.R. (1991), 'Facilitating the development of knowledge based systems', *AICOM* 4(2/3), 60-73.

Nwana, H.S., Shave, M.J.R. Bench - Capon, T.J.M. & Paton, R.C. (1992), Domain - driven knowledge modelling: Mediating and intermediate representation for knowledge acquisition, in 'Proceedings of 6th European Knowledge acquisition Workshop, Heidelberg and Kaiserslautern, Germany', pp. 251-263.

Nystedt, L. (1983), *The Situtation: a constructivist approach.*, New York: Academic Press, pp.94-112. In J.A. Adams - Webber and J.C. Mancuso (Eds): Applications of Personal Construct Theory.

Paton, R., Nwana, H.S. Shave, M.J.R. & Bench - Capon, T.J.M. (1991), From real world problems to domain characterisations, in 'the Proceedings of the 5 the European Knowledge Acquisition Workshop (EKAW _ (I) , Crieff, Scotland'.

Quillian, M.R. (1968), *Semantic Memory*, MIT Press, pp. 216 - 270. In Semantic Information Processing Edited by Marvin Minsky.

Quinlan, J.R. (1983), *Learning Efficient Classification Procedures and their application to chess and games*, New York: Tioga. In Michalski R.S. Carbonell J.G.and Mitchell T.M. (Eds), Machine Learning: An artificial intelligence approach.

Rappaport, A. T. & Gaines, B.R. (1990), 'Integrated knowledge base bulding environment', *Knowledge Acquisition* 2, 51-71.

Recogzei, S. & Plantinga, E. P. O. (1987), ' Creating the domain of Discourse: Ontology and inventory', *International Journal of Man- Machine Studies* 27, 235-250.

Rumelhart, D. E. & McClelland, J. L. (1986), *Parallel Distributed Processing, Explanation in the Microstructure of Cognition*, Vol. 1, The MIT Press.

Schreiber, G. (1993), *A KADS Domain Description Language*, Academic Press, Harcourt Brace Javanovich Publishers, pp. 72-91. In Guss Schreiber, Bob Wielinga & Joost Breuker (Eds): KADS A Principled Approach to Knowledge - Based System Development.

Schreiber, G., Wielinga, B. & Breuker, J. (1993a), *Introduction and Overview*, Academic Press. Harcourt Brace Jovnnovich Publishers pp. In Guss Schreiber, Bob Wielinga & Joost Breuker (Eds.): KADS A Principled Approach to Knowledge - Based System Development.

Shaw, M.L.G. & Gaines, B.R. (1987), 'Techniques for knowledge acquisition and transfer', *International Journal of Man Machine Studies* 27, 251-280.

Shaw, M.L. & Woodward, J.B. (1990), ' Modelling Expert knowledge', *Knowledge Acquisition* 2 , 179-206.

Sowa, J.F. (1984), *Conceptual Structures: Information Processing in Mind and Machine*, Addison - Wesley Publishing Company.

Steel, L. (1990), 'Components of expertise', *AI Magazine, Summer issue* pp. 28 -49.

Vanwelkenhuysen, J. & Rademakers, p. (1992), *Mapping Knowledge - level Analysis on to a computational Framework*, London: Pitman, pp. 681-686. In L. Aiello (ed), Proceedings ECAI'90, Stockholm.

Wielemaker, J. & Anjewierden, A. (1994), *XPCE Reference Manual*, University of Amsterdam.

Wieinga, J. & de Velde, V., Schreiber, G. & Akkermans, H. (1993), *Towards a Unification of Knowledge Modelling Approaches*, Springer - Verlag, Berlin, pp. 299-335 In : J.M. David and J.P. Krivine and R. Simmons (Eds): Second Generation Expert Systems.

Wielinga, B., Schreiber, T. & Breuker, J. (1992), 'KADS: A modelling approach to knowledge engineering', *Knowledge Acquisition* 4, 5-53.

Woodward, B. (1989), 'Integrating task demands and problem - solving methods to develop useful taxonomies for knowledge acquisition'. In the Proceedings of the Third European Workshop on Knowledge Acquisition for Knowledge - Based Systems EKAA' 89, Paris, France (June).

Woodward. B. (1990), 'Knowledge acquisition at the front end: Defining domain', *Knowledge Acquisition* 2, 73-94.

Yourdon, E. (1989), *Modern Structured Analysis*, Prentice - Hall International, Inc.