The KADESS Knowledge-Based System: Employing the KADS methodology in an engineering application

Sarah Price
Dept of Artificial Intelligence
University of Edinburgh

John Kingston
AIAI
University of Edinburgh

AIAI-PR-55

August 1993

This paper was presented at the Sixth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Edinburgh, 1-4 June 1993

Artificial Intelligence Applications Institute
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom

ABSTRACT

There has been a growing desire within recent years for a more formalised approach to KBS development. This paper describes and justifies the use of the *KADS* methodology, the most influential such formalism to date, in guiding the development of a small knowledge-based system (hereafter KBS) in an engineering application. The KADESS system [1] concerns itself with the checks that a practicing civil engineer or architect would conduct on a proposed industrial building to ensure that all safety and stability criteria are met.

The KADS methodology regards knowledge engineering essentially as a modelling process. This paper traces the progress of this model-based approach, where the initial real-world expertise is gradually transformed into system implementation.

INTRODUCTION TO THE DOMAIN

The specific types of industrial building considered by KADESS are *portal frames* - a very common structure comprising 90% of all single-storey buildings found on modern industrial estates. The reader can immediately visualise a portal frame building by recalling the appearance of a typical example: a *B & Q* or *Texas Homecare* store. These buildings are favoured in industry due to their relatively simple and cheap construction and their comparatively low roofs, leading to low heating and energy costs.

Though a portal frame might represent one of the simplest forms of three-dimensional frameworks, an engineer engaged in their design needs as sound a knowledge of principles of structural design as would be required in any more elaborate project. The checks that a civil engineer makes on a proposed portal frame building may be subcategorised into *ultimate limit states* and *serviceability limit states*.

In general, *ultimate limit states* refer to the overall *safety* of a structure. A safe building, in this context, means a building which will not collapse under the anticipated working loads.

Safety factors are built in to these calculations so that a building is actually safe to some specified degree over and above normal working load; the precise safety factor to be applied is fully specified in the British standards.

Serviceability limit states, on the other hand, would not be expected to result in the ultimate collapse of the building if the engineer made errors at the design stage. This area of design concerns the *use* but not necessarily the *strength* of a building. If a structure is made of concrete, serviceability checks might be performed to ensure that undue tensional forces do not cause the material to crack; if a structure is fairly lightweight, serviceability checks might be necessary to ensure that vibration levels are not so great that they cause discomfort to the occupants of the building. Thus, serviceability problems may cause a building to become unusable or uninhabitable whilst being, at the same time, perfectly safe.

A possibly apocryphal story which is quoted by architects in illustration of this point is the case of the World Trade Centre in New York. The story goes that the top ten floors cannot be let since the building sways too much in high wind - an effect which would obviously become progressively more exaggerated as one rises to higher and higher levels within the building. Likely reactions from an occupant might range from inconvenience to discomfort to rigid alarm, but would not with justification include doubts on the ultimate strength and stability of the building.

Of all the different and complex structural criteria to be considered by the practicing buiding designer, the KADESS system focuses on the *ultimate limit states* described above.

EXPERT SYSTEMS IN ENGINEERING APPLICATIONS

There is a recognised key set of problem *types* which arise in engineering that expert systems have been used for. For the sake of brevity, one example only of each of the generic types appears in the following list:

- Interpretation: PROSPECTOR for interpreting geological surface data to evaluate potential mineral deposits [Campbell et al, 1982]
- Monitoring: PLATFORM for monitoring the status of ongoing construction projects [Levitt, 1985]
- Planning: ISIS for planning work flow in a job shop scheduling environment [Fox, 1984]
- Design: HI-RISE for the preliminary design of high-rise buildings [Maher, 1984]

[1] The acronym KADESS stands for the *KADS Approach to the Design of Steel Structures*

- Diagnosis: SPERIL-1 and SPERIL-II for diagnosing possible causes of structural failure [Ishizuka, 1981] [Ogawa, 1984]

All of these kinds of problems occur somewhere in most domains. The specific area which concerns KADESS is a subclass of the *interpretation* problem type, ie., *assessment*. KADESS's function is to *assess* a candidate portal frame design against the relevant statutory codes.

The effectiveness of knowledge-based expert systems in engineering has been discussed extensively elsewhere, [2] but, in summary, their strength lies in being able to provide help with certain problem types where standard algorithmic programming techniques fail. For example, KADESS exploits A.I. techniques in order to capture the expertise of the practicing human engineer in this domain. Even the knowledge acquisition process itself is an A.I. technique which can serve to deepen and sharpen the experts' own understanding of the knowledge. In addition, the standard labour-saving benefits of computational systems also apply.

THE KADS APPROACH TO THIS APPLICATION

Since it is becoming more and more widely accepted within industry that expert systems have much to offer, it is imperative that KBS designers can use a methodology for enabling them to build systems quickly, understandably and cheaply. An unstructured prototyping approach can often lead to errors in the final system which can be very difficult to track down and understand. KADS is such a KBS development methodology which aims to enable system designers to "look before they leap". The intention of KADS is that it should lead to accurate and clear documentation of the various stages of system design, leading to reduced errors in the final system.

WHAT IS KADS?

Knowledge-based system development, according to the KADS methodology, is essentially a modelling activity. The models are simply formalised descriptions of observed behaviour, or behaviour whose description is derived through knowledge elicitation. The underlying intention of the methodology is that it should provide a *guiding framework* for KBS development rather than a rigid structure which must be blindly adhered to. Thus, the developers of the methodology are keen to stress that the general approach is *normative* rather than *prescriptive*, meaning that it provides guidance on *what* should be produced, rather than *how*.

The KADS approach has been to take a traditional software engineering approach to KBS development. This means that a rigorous and disciplined analysis and requirements stage is undertaken first - well before any design decisions are made. Prototypes may be used, therefore, in an experimental way, to guide the choice of an appropriate AI method or possible user interface; the intention is not that the prototype represents a development paradigm.

Further information on KADS can be found in documentation from the current ESPRIT KADS-II project (e.g. [Porter, 1992]) or in [Hickman et al, 1989]

PRAGMATIC KADS

Pragmatic KADS is a variation on the original KADS methodology intended for use on smaller KBS projects. The variation was first introduced by one of the authors, in the paper *Pragmatic KADS: A Methodological Approach to a Small Knowledge-Based System Project*, (see [Kingston, 1992a]). The need for *Pragmatic KADS* was perceived in response to the criticism of KADS that excessive time is required both to construct the detailed models and to write the large number of reports required to document progress.

A STEP BY STEP APPROACH

Having introduced the origins and objectives of the methodology, this section focuses more closely on the order of events recommended by KADS in the construction of this knowledge-based system

1. **Task selection**. This step commences with the investigation of a company to see whether it might benefit from the assistance of a knowledge-based system in any of its activities. The KADS feature which is used to guide this initial step is called the *model of co-operation*, which is a model of the tasks themselves. The construction of this model involves three stages:

    - creating a breakdown of each sub-task which contributes to the overall task, with additional information on who does what;
    - identifying check dependency information;
    - identifying the advantages or feasibility of applying a KBS in certain areas, depending on which tasks would be performed by the system alone, which by the user, and which by the two operating in tandem.

    In the case of KADESS, a *model of co-operation* was not required as the choice of application area

[2] See, for example, Dym & Levitt, 1991, *Knowledge-Based Systems in Engineering*, pp 4 - 19.

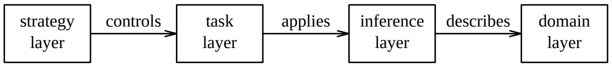

Figure 1: The interactions between layers within the KADS *model of expertise*

had already been selected by the Artificial Intelligence Applications Institute at the University of Edinburgh.

2. **Knowledge acquisition**. The knowledge acquired can come from any relevant source, but is mainly via knowledge elicitation interviews with experts in the field, or through published works.

3. **Knowledge analysis**. Acquired knowledge, from whatever source is analysed here. Understanding or analysing the nature of expertise is clearly one of the major objectives and challenges to the designer of a KBS and the methodology facilitates the capture of its exact nature and structure via the *four layer model of expertise.* This central feature of KADS was greatly influenced by the initial work of Brachman, 1979 [3], and thereafter of Clancey, 1983 [4]. Brachman was the first to identify a set of epistemological primitives, and Clancey built on this early work by further identifying three different types of knowledge: *strategic, structural* and *support* knowledge. In 1984, Breuker and Wielinga [5] combined Brachman and Clancey's work and succeeded in identifying a minimum set of basic elements for describing expertise: *objects, knowledge sources, models, structures*, and *strategies.* Out of these components developed the KADS model of expertise. The model has four layers which are identified in figure 1 and described in the text that follows:

   The *model of expertise* is situated at an abstract level away from the real world and from any matters to do with implementation. Its function is to assist the designer in three major ways:

   - It guides the interpretation and analysis of data;
   - It provides a framework for categorising knowledge according to the role it plays in the reasoning process;
   - It provides a model of the steps which are taken in the reasoning process.

The expertise primitives which have been developed within KADS and which together comprise the four-layer model of expertise may be outlined as follows:

(a) **The domain layer** - the static concepts, relations and structures specific to the domain; the raw data.

(b) **The inference layer** - the layer in which knowledge is contained of the different types of inferences that can be made, laid out in a declarative manner. This component of the model of expertise is known as the inference structure. An initial objective to be realised at this level is to define exactly what type of task is being undertaken by the KBS, eg. assessment, configuration, diagnosis, etc. The KADS methodology supports the design process by providing a *library* of "generic" models spanning the range of possible task classifications. Once a broad task definition has been agreed, an appropriate *interpretation model* is chosen from the KADS library of models and is adapted or *instantiated* as necessary so that it fits in well with the problem domain. When instantiated to the domain, the interpretation model is referred to as the *inference structure.*

    The interpretation model is a central design feature of the four-layer model of expertise; its function is to enable the knowledge engineer to interpret and organise the data, by acting as a template, guiding knowledge acquisition and analysis. In utilising this feature the KADS methodology may be described as a top-down approach of incremental *model refinement*; the models are taken *off the peg* and, though a few adjustments may be necessary here and there to get the chosen model to fit more exactly into the domain, the framework is basically there upon which to hang future development and design decisions. This contrasts with the bottom-up approach of *model creation*, where a new *tailor-made* model is built from scratch.

(c) **The task layer** - Whereas the previous layer described the different types of inferences that can be made in a declarative manner, the task layer provides a procedural in-

[3] Brachman, R. *On the epistemological status of semantic networks* in Findler, N (ed), *Associative Networks*, Academic Press, New York, 1979

[4] Clancey, W. *The epistemology of a rule-based expert system - a framework for explanation*, Artificial Intelligence, 20, pp215-251, 1983

[5] Breuker, J., Wielinga B., *Models of Expertise in knowledge acquisition* in Guida, G., Tasso, C. (eds), *Topics in Expert System Design: methodologies and tools*, North Holland Publishing Co, Amsterdam, 1988

terpretation. This component of the four-layer model provides a procedural ordering of the tasks to be carried out by the KBS.

   (d) **The strategy layer** - Monitoring of the problem-solving process takes place at this level so that conflict resolution may be performed in the event of a deadlock. If necessary, this may involve choosing an alternative task structure. A strategy layer is necessary in the event of highly complex problem solving systems and many KBS projects do not require this level of analysis.

4. **Design**. System development descends from the above high point of abstraction to an intermediate level. KADS moves closer to conventional software engineering techniques here by recommending the subdivision of the design phase into the three phases of *functional, behavioural and physical design*. Decisions are made at this stage which pave the way for implementation.

5. **Implementation**. Under the influence of the decisions made and information gained at the previous analysis and design stages, the code is finally written in the language or toolkit selected by the designer as most appropriate for the KBS.

The following section examines more closely how the KADESS system was built on the foundations recommended by KADS.

ANALYSIS: THE KADS MODEL OF EXPERTISE

The KADS *model of expertise* represents a fundamental building block of the analysis phase of system development. Each of its four layers will now be considered in turn with specific reference to the domain of portal frame buildings.

The Domain layer

Considerable knowledge acquisition and analysis were required for identifying the static concepts, relations, and structures which make up the domain layer. The underlying principles and values upon which a safe and stable portal frame building must be constructed are found in the voluminous British standards document BS5950, part 1. It is necessary to interpret this, however, and the architects and civil engineers who kindly gave of their time for this project helped identify the essential checks which practicing engineers perform on proposed portal frames. The domain layer of the four-layer model consists of, firstly, identifying and, secondly, describing the relevant checks. A document outlining each of the checks was prepared at this stage with entries for each one as in the following extract for selecting universal beams:

- **Step**: Universal beam selection

  Description: Calculating the plastic modulus and selecting a suitable universal beam (UB)
  Input:The figure for the full plastic moment (Mp) as calculated in checks for *possible collapse mechanisms.*
  Dimensions of the UB section which it is proposed to use, obtained from *A Check List for Designers*, the Steel Construction Institute, September, 1986.
  A figure for the design strength (Py) of steel to BS4360, obtained from BS5950, table 6.
  Two figures obtained from the plastic class of section in table 7, BS5950:
  1. figure quoted for the outstand element of compression flanges for rolled sections;
  2. figure quoted for webs with neutral axes at mid-depth.
  Output:The system suggests, on the basis of its calculations, whether the degree of plasticity required is less than the degree of plasticity provided.

THE INFERENCE LAYER

This layer contains knowledge of the different types of *inference* that can be made from the domain layer above. It is here also that a decision is made on the most appropriate task-type for the KBS, eg. whether *diagnosis, heuristic classification, prediction,* etc. KADS offers much guidance during the analysis phase of development by providing a library of "generic" problem-solving tasks, each one of which is referred to as an *interpretation model.*[6]. The function of these models is to serve as templates for structuring and obtaining knowledge.

It is essential that only the most *appropriate* interpretation model is chosen from the KADS taxonomy since this early decision acts as the foundation for much of the subsequent work; the model is the starting point for data analysis. After deliberation, it was decided that the most suitable task-type for the KADESS Project was that of *assessment.* Out of the range of possible task-types, the reason for the selection of the *assessment* model rests on the intention that the system should be able to offer grades of acceptability by which a proposed design conforms with the regulations. In answer to the question "Does

[6] A full discussion on the KADS model library, the range of *interpretation models* and the task-types supported by the methodology can be found in chapter 5 of [Hickman et al, 1989]

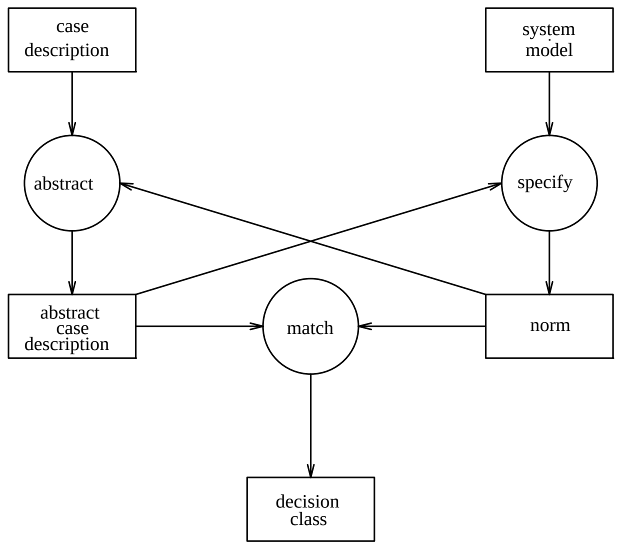

Figure 2: The Interpretation Model for Assessment

Building Plan X conform with all necessary design criteria contained in BS5950", it was anticipated that a fuller response than just "Yes" or "No" would be desirable. The choice of the interpretation model for *assessment* would provide KADESS with a framework for making desirable distinctions on exactly how well the building conforms with the regulations.

The KADS interpretation model for assessment is shown in figure 2 The two basic components of the KADS interpretation models are boxes and ovals: the boxes represent *knowledge roles*, or roles that the domain elements may occupy during the inference process, and the ovals represent *inference actions*, or the inferences that are performed on these domain elements. The meanings of each component are as follows:

- *the case description* represents the design that the engineer is working on. This will not be a static description; elements of this knowledge role will change as the design changes or becomes refined in the light of information gained from BS5950. Additionally, since the various sections of an industrial building must be checked individually as well as collectively in the case of a whole design, the case description will vary according to which part of the overall design is under consideration. But regardless of whether the portal frame designer is considering a complete or a partial design, the case description represents everything that he or she knows about the design; it will contain both features that will need and that will not need to be checked against the British standards.
- *the abstract case description.* This knowledge role is obtained by performing the inference action *abstract* on the *case description.* Only those features that are relevant to the checks that need to be performed in accordance with the regulations are abstracted from the case description to produce this knowledge role. All superfluous information, in the context of the present task, is disregarded.
- *the system model.* One way to visualise the system model would be to imagine the exact design contained in the case description as a thoroughly checked and ideal example: a perfect design in every respect according to the regulations. This clearly is not possible in reality since even within one design type, features vary infinitely and a system could not contain ideal examples of all the designs necessary to make this possible. An alternative is to view BS5950 itself as the idealised and perfect version of the design. This is a reasonable approach since if it can be said that all features contained within the abstract case description match favourably against the British standard, then the design is complete.
- *the norm.* If the system model is BS5950, then clearly some kind of selection process must be performed so that an appropriate subset of standards may be applied to the design so far. This selection is executed by the *specify* inference action.
- *the decision class* represents the final verdict on the adequacy (or otherwise) of the building design with respect to the regulations. The decision is reached by *matching* the components of the *abstract case description* with the *norm* described above.

In summary, the model outlines the execution of an assessment task in the following way:

- *abstracting* from the *case description* the relevant feature(s) to comprise an *abstract case description.*
- *specifying* from the *system model* the feature(s) which make up the *norm.*
- *match*ing the *norm* to the *case description* to produce the *decision class.*

The following section makes this abstract discussion more concrete by showing how the KADS interpretation model for assessment is *instantiated* to the domain. This instantiated template is known in the methodology as the *inference structure.*

INFERENCE STRUCTURE

The instantiated interpretation model is referred to in KADS as the *inference structure.* Figure 3 presents the KADS modelling process in diagrammatic form and shows the flow of development from interpretation model selection to final instantiation within the domain.

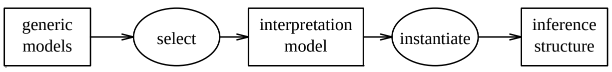

Figure 3: The modelling process in KADS - from interpretation model selection to instantiation of the inference structure

It is often the case that interpretation models have to be adapted or customised to fit more exactly the problem domain. This was found to be the case with the KADESS system. The adapted and instantiated version of the assessment model is shown in figure 4.[7] A breakdown of the flow of action in this model may be made in the following way:

1. Select one check to perform from a list of possible checks which may be made to ensure the stability of a portal frame.

2. Perform check dependency analysis on the selected check to ensure that any necessary preconditions are satisfied.

3. Select from the *key design features* those items of information necessary for the execution of the check.

4. Select from the *British standards relevant information* those items of information necessary for the execution of the check.

5. Compare the information obtained from item 3 above with the information obtained from item 4.

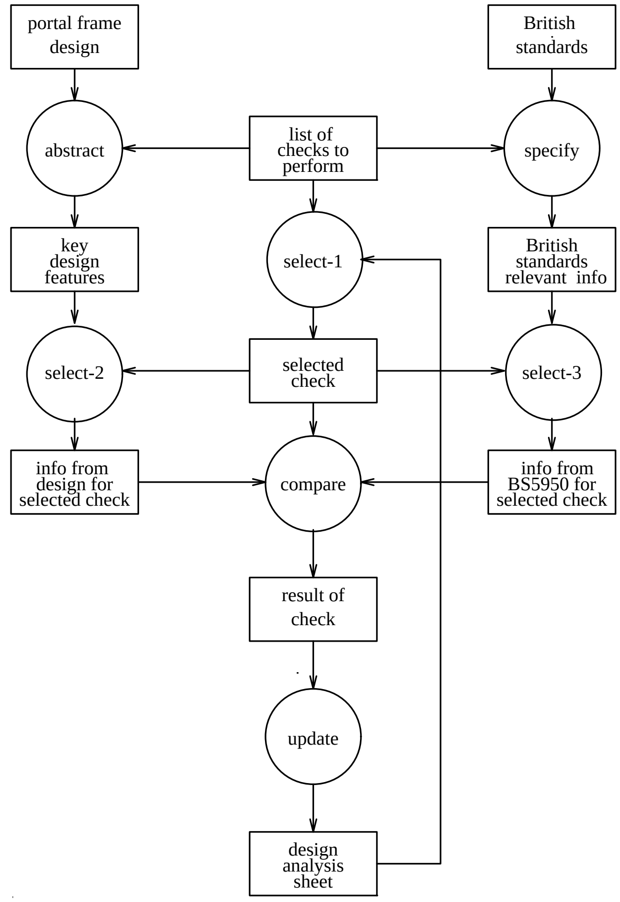

Figure 4: Inference structure for the KADESS system

6. Produce a check result and update the *design analysis sheet* with this result.

An obvious change from the KADS interpretation model for assessment has been to substitute the knowledge role labels with labels that are more meaningful within the domain of portal frame buildings. Thus, the knowledge role *case description* becomes *portal frame design* and *system model* becomes *British standards* and so on.

The inclusion of the two new inference actions *select-2* and *select-3* has been made in recognition of the complexity of the domain: a large amount of information is stored both about the specific building to be checked and about general principles of design and requirements of materials, as detailed in British standards BS5950. From this mass of information it is necessary to *select* the precise portions necessary to perform the assessment. Thus *select-2* and *select-3* were devised to make this action explicit. (The numerical suffixes do not indicate that two or three items are being selected

[7] It should be noted that a pure interpretation of the KADS methodology would not authorise the inclusion of the *update* inference action and the *design analysis sheet* knowledge role in the inference structure shown in figure 4 Strictly, these are follow-up actions rather than inferences and the methodology stipulates that only the latter should appear in the *inference structure*. Thus, the inclusion of these "illegal" categories are allowable only within the framework of *Pragmatic KADS*; they were found to be most helpful in analysing at a high level how the system would function and what was expected of it.

here; their purpose is to differentiate the two otherwise identical inference actions in subsequent analysis.)

The expansion of the central column of inference actions and knowledge roles is an important modification to the KADS interpretation model. Once again, this modification has been necessary due to the type and abundance of information contained mostly within the British standards. The latter cannot possibly consist of a replica of any possible portal frame design, identical in every respect except that any structural defects are corrected. What it *does* consist of is a complexity of tables, formulae and diagrams. Thus a key design feature from the portal frame plans cannot be straightforwardly compared with a corresponding feature from BS5950; assessment cannot proceed without an intermediate inference action (*select-1*) which selects from both sides only the information necessary for the selected check which it is desired to perform. The *list of checks to perform* consists of the dozen steps outlined in the *domain layer* of the preceding section. *select-1* acts upon these to choose a *selected check*.

Finally, the *decision class* knowledge role has been replaced by the more expressive *result of check* and is followed by the *update* inference action and *design analysis sheet* knowledge role. The latter has been added in recognition of the fact that there are several checks which may be performed on a portal frame design and that the user of the KADESS system would certainly wish to be able to contextualise the assessment information by being able to see a resumé of all the check results in a coherent format. The design analysis sheet is updated upon the conclusion of each check and the system either halts at that point or, at the user's instigation, continues by selecting another feature to check. This is indicated by the long arrow linking the *design analysis sheet* with the *select-1* inference action above.

Now that the KADS template or *interpretation model* has been instantiated to the domain, the methodology requires the system designer to produce a report in which the elements identified in the *inference structure* of figure 4 are individually described. Entries for one example *knowledge role* and one example *inference action* appear below:

Knowledge Role : **list of checks to perform**

Description : a list of all checks which it is necessary to make to ensure the structural stability of a portal frame building.

Created by : initial knowledge acquisition.

Inference action : **update**

Description : update the design analysis sheet with the result of the check last performed.

Input : result of check

Output : updated design analysis sheet

The task layer

The task layer of the model of expertise is provided by the *task structure*. The function of this structure is simply to outline the tasks which are to be performed and to impose an ordering upon them. This layer also makes explicit ordering information concerning check dependencies. For example, if the check for the suitability of a selected universal beam requires as input a figure relating to the full plastic moment as calculated in a previous step, then the check cannot be performed unless the preliminary step(s) has been taken.

STRATEGY LEVEL

The strategy level is essential for decision making in complex problem-solving systems where there is likely to be a deadlock or conflict in the ordering of tasks. In the KADESS project, the checking procedures are complex in themselves, but according to how an experienced engineer would approach the problem, the various stages of checking are systematically outlined and executed. Therefore, no conflict resolution is necessary and a strategy level analysis is not needed.

THE DESIGN PHASE

KADS has rather less to say on the design phase of system development than on the analysis phase, for which extensive guidance is found in the methodology. However, the basic starting-point of design is that it should preserve the structure of the *model of expertise* and particularly of the *inference structure*. It is important that this should be so since these outputs of the analysis phase were created with the specific intention of enabling a smooth-running and effective design phase.

The subdivisions of the design phase may be itemised as follows:

- functional design
- behavioural design
- physical design

This subdivision is comparable to that found within conventional software design methodologies.

Functional design

The *functional design* involves performing a functional decomposition. The point of the functional decomposition is that acts as an aid to the KBS designer in ensuring that no desired function is omitted. It further serves to make explicit the links and interactions between the various components making up the system. However, the resulting diagram is almost hopelessly complicated.

BEHAVIOURAL DESIGN

Behavioural design is essentially choosing the most appropriate A.I. techniques for use by the system. It will be immediately apparent that KADS current paucity of guidance on design matters cannot logically be supplemented by techniques borrowed from conventional software engineering. Therefore, the KADESS project has relied upon the "probing questions" approach, originally proposed by Kline & Dolins [Kline & Dolins, 1989] and developed further within the Artificial Intelligence Applications Institute. The result of this approach led to the design conclusion that KADESS would benefit most from an object-based representation, consisting of objects, frames, facts, schemas, and rules. Therefore, the tool chosen for the job would ideally possess all these features. The specialised rule-based, KBS development environment chosen was the toolkit ART-IM (Automated Reasoning Tool for Information Management).

PHYSICAL DESIGN

The *physical design* for the KADESS system involved further report writing in the shape of exact descriptions for each of the elements identified during the *functional decomposition*. Each description was supplemented with details of how each element was to be implemented in the chosen toolkit, ART-IM. An example extract from the report produced at the *physical design* stage follows:

- **portal frame design**: This information is conveyed to KADESS by means of ART-IM's ability to read in from an external file. The user provides the information for the external file from the detailed drawings and portal frame design that they have before them. It is envisaged that KADESS will guide the user on which information is required by means of the interface. The user's responses are sent to the external file and, when complete, the values contained therein are read in by KADESS as required for use in its assessment steps.

- **info from BS5950 for selected check**: the information necessary from BS5950 for performing the assessment for a specific selected check. This information appears for the most part as tables within BS5950, and in line with the recommendation of the *probing questions* analysis, is best represented within ART-IM as *facts* or *schemas*.

- **dependency information about checks**: Represented by rules and facts. The rules indicate whether any preconditions must be met prior to a check being performed, eg. it is necessary to perform step X before step Y, etc.

THE IMPLEMENTATION PHASE

KADS has little guidance to offer on implementation matters. The current emphasis of the methodology is on the analysis phase of system development. Though this will almost certainly change with the publication and marketing of the improved methodology under KADS-II, at present, implementation is seen largely as a programming exercise for which the usual software engineering techniques are applicable.

As the design phase of development was built upon the foundations laid in the analysis phase, so also the implementation phase builds upon the decisions made and issues raised in the preceding design phase. The selected toolkit ART-IM contains all the features identified as useful during the *behavioural design* and the resultant system was finally implemented in accordance with the intentions expressed during the *physical design*.

CONCLUSIONS

In the final analysis, KADS provides useful guidance to the knowledge-based system designer, but not without some cost in terms of high administrative overheads (report-writing and model designing). The methodology encourages a strong *look before you leap* strategy to KBS development. The criticism that one might make in connection with the administrative overheads is that to follow the methodology religiously can mean *looking* a little too long at the problem before *leaping* into implementation. Less report-writing would free up more time for code-writing. This criticism is most relevant to small and medium-scale systems which are typically least at risk from *ad hoc* KBS development procedures. That is why this particular KBS must be described more strictly as conforming to *Pragmatic KADS*, [Kingston, 1992a], as described in section 2.4.

The hypothesis tested in this project was that KADS was able to guide the development of a knowledge-based system whose objective was to reasonably and properly represent the British standards, and to perform assessment steps appropriate to the design of a portal frame building. Both the application of the KADS methodology and of the "probing analysis" approach to design worked well in the to this end.

In conclusion, the construction of the KADESS is a good prototype for demonstrating how a KBS can be used in portal frame design assessments and potentially other similar engineering applications; secondly, the system provides a base for a tutoring or demonstration system for junior engineers; finally, it is hoped that the results of this research will provides clear guidance to other researchers, KBS designers and students alike into the KADS methodology itself.

References

[Brachman, 1979] Brachman, R. *On the epistemological status of semantic networks* in Findler, N (ed), *Associative Networks*, Academic Press, New York, 1979.

[Breuker, 1988] Breuker, J., Wielinga B., *Models of Expertise in knowledge acquisition* in Guida, G., Tasso, C. (eds), *Topics in Expert System Design: methodologies and tools*, North Holland Publishing Co, Amsterdam, 1988.

[Campbell et al, 1982] Campbell, A.N., et al, *Recognition of a hidden mineral deposit by an artificial intelligence program*, Science 217 (3), 1982.

[Clancey, 1983] Clancey, W. *The epistemology of a rule-based expert system - a framework for explanation*, Artificial Intelligence, 20, pp215-251, 1983.

[Fox, 1984] Fox,M.S., Smith,S.F., *ISIS - A Knowledge-based system for factory scheduling*, Expert Systems 1 (1), 1984.

[Hickman et al, 1989] Hickman, F., et al, *Analysis for Knowledge-Based Systems: a practical introduction to the KADS methodology*, Ellis Horwood, Chichester, 1989.

[Ishizuka, 1981] Ishizuka, M., Hutchinson, E.L., Weitzman, L., *SPERIL-1: Computer-Based Structural Damage Assessment System*, Report CE-STR-81-36, School of Civil Engineering, Purdue University, Lafayette, IN, 1981.

[Kingston, 1992a] Kingston, J., *Pragmatic KADS: A Methodological approach to a small knowledge-based system project*, Expert Systems, 4, 4, November 1992.

[Kline & Dolins, 1989] Kline,P. & Dolins, S., *Designing Expert Systems: A guide to selecting implementation techniques*, John Wiley & Sons, 1989.

[Levitt, 1985] Levitt, R., Kunz, J.C., *Using knowledge of construction and project management for automated schedule updating*, Project Management Journal 16 (5), 1985.

[Maher, 1984] Maher, M.L., *HI-RISE: A Knowledge-based expert system for the preliminary design of high rise buildings*, Ph.D. Dissertation, Pept of Civil Engineering, Carnegie Mellon University, Pittsburgh, PA, 1984.

[Ogawa, 1984] Ogawa, H., Fu, K.S., Yao, J.P., *An expert system for damage assessment of existing structures*, Proceedings of the First Conference on Artificial Intelligence Applications, IEEE Computer Society, 1984.

[Porter, 1992] Porter,D., *KADS-II: Towards the CommonKADS method*, ESPRIT Project P5248 KADS-II, KADS-II/T5.1.2/PP/TRMC/001/1.0. Touche Ross & Co