ELSEVIER

Journal of Materials Processing Technology 107 (2000) 384–389

Journal of Materials Processing Technology

www.elsevier.com/locate/jmatprotec

Knowledge-based engineering for SMEs — a methodology

P.J. Lovett*, A. Ingram, C.N. Bancroft

Knowledge Engineering and Management Centre, School of Engineering, Coventry University, Priory Street, Coventry CV1 5FB, UK

Abstract

This document describes a Knowledge-Based Engineering (KBE) project currently in progress at Coventry University. The paper firstly explains the underlying mechanisms of KBE, and the significant benefits that can be achieved from their application. These are illustrated using industrial examples. The need for a methodology for KBE system development is examined, as are the differing requirements in this respect of small and large organisations. Existing methodologies are discussed, and the aims and approach for a new methodology, specifically aimed at SMEs (small- to medium-sized enterprises — those having less than 250 employees), are outlined. Work on this methodology is underway, involving extensive collaboration with local enterprises. An overall framework for the activities to be performed has been drawn up, and an integrated modelling approach has been devised. The paper ends with a discussion of the ways in which the special needs of SMEs are being fulfilled by the methodology.

Keywords: Knowledge-based engineering; Knowledge; Methodology

1. Introduction

A research and development project known as REFIT (revitalisation of expertise in foundries using information technology) [1] is being carried out at the Knowledge-Knowledge Engineering and Management (KEM) Centre at Coventry University. It aims to improve the competitiveness of SMEs in the foundry industry. However, although the work is specifically targeted at the needs of foundries (as it is funded for this purpose under a European Social Fund initiative — ADAPT 1999), the principles and practices described will apply equally to many other engineering sectors. The paper is particularly concerned with the construction of a methodology for the development of KBE systems within SMEs.

2. KBE

A knowledge-based system (KBS) is the one that captures the expertise of individuals within a particular field (the "domain"), and incorporates it and makes it available within a computerised application. The level of complexity of the tasks performed by such a system can vary greatly. However, it can generally be said that while a domain expert would find them routine, they would be outside the capabilities of a person unfamiliar with the domain.

A KBE application is further specialised, and typically has the following components — geometry, configuration, and engineering knowledge:

- Geometry — there is very often a substantial element of computer-aided design (CAD). Most of the software used to create KBE applications either has CAD capabilities built in, or is able to integrate closely with a CAD package.
- Configuration — this refers to the matching of valid combinations of components.
- Engineering knowledge — this enables manufacturing and other considerations to be built into the product design.

When a candidate application area requires a high degree of integration of the above elements, KBE is likely to be the best method for its integration. KBE is sometimes termed rule-based engineering, as within the discipline knowledge is often represented by rules. These may be mathematical formulae or conditional statements, and although simple in concept, they may be combined to form complex and powerful expressions. Some example rules in the above categories follow:

- Geometry:

$$\text{Cylinder\_1\_Position} = \text{Cylinder\_2\_Position} + \text{Vector}(2, 4, 12)$$

* Corresponding author. Tel.: +44-2476-8999; fax: +44-2476-8604.
E-mail address: p.lovett@coventry.ac.uk (P.J. Lovett).

0924-0136/00/$ – see front matter
PII: S0924-0136(00)00728-7

Rules of this type enable changes that are made to an individual element of a CAD drawing to be reflected throughout the rest of the drawing. This feature, which is known as "parametric modelling", eliminates a large proportion of the repetitive tasks involved in producing a design.

- Configuration:

$$\text{IF Wheel\_1\_Diameter} > 5 \text{ THEN Number\_Of\_Spokes} > 12$$

Rules of this type describe conditions that must be observed for configurations of components to be "legal".

- Engineering knowledge:

$$\text{IF Material} = \text{Aluminium\_2 THEN Min\_Thickness} = 0.5$$

Such a rule ensures that manufacturing capabilities are taken into account at the design stage of product development.

The following example of a typical KBE application demonstrates some of the considerable benefits to be gained from its use: a "black box" application for the design of a class of products can be built, which as well as having CAD capabilities, can incorporate engineering knowledge, such as that relating to material properties, permitted stress levels, manufacturing feasibility, manufacturing and material costs, and so on. The designer is then able to exercise his creativity in designing the product, within the constraints imposed by the embedded knowledge. This will prevent his design from being rejected on the grounds that the product cannot be manufactured, or because it is too expensive to manufacture, or because it does not have the required physical properties.

The main advantage of the KBE approach, however, is the compression of lead times through the automation of repetitive procedures. Key employees are then freed to concentrate on more creative and cost-effective activities. Moreover, KBE applications are not limited to the design area, and include product configurators for use as sales aids, and costing software to provide estimates and quotations based on company-wide knowledge.

It is important to appreciate the difference between KBE systems, as described above, and the more generalised KBSs of which they are a subset. The latter are often called expert systems, and utilise a centralised store of knowledge — in much the same way as a database makes use of a central store of data. KBSs have been developed for a broad range of application areas, and may embrace more sophisticated types of knowledge. KBE systems, on the other hand, are usually provided with specialised geometrical capabilities, and the ability to embed engineering knowledge within a product model.

Major companies already realising significant benefits from the use of KBE include the following:

- *Lotus engineering*. This used the integrated car engineer (ICE) system in the design of the Lotus Elise. ICE consists of a vehicle layout system, and modules to support the design of suspension, engines, powertrain, wheel envelope and wipers [2].
- *The Boeing Commercial Airplane Group*. This uses KBE as a tool to capture airplane knowledge to reduce the resources required for producing a design [3].
- *Jaguar cars*. The company's KBE group devised a system that reduced the time taken to design an inner bonnet from 8 weeks to 20 min [4].

3. The need for a methodology

A methodology is essentially a set of instructions and guidelines on how to perform a complex procedure. It details the individual sub-tasks, how they should be carried out, in what order, and how the work should be documented. It is possible to develop a KBE (or any other) application without availing oneself of a methodology. Furthermore, it might be thought that the time and effort required to do so would be better spent developing the application itself. However, the use of a methodology is not simply beneficial — it is vital for the quality, reusability and maintainability of the delivered system. As systems requirements change, new solutions tend to evolve from existing ones, so computer applications and their descendants can outlive the personnel involved in their initial development. Some of the problems currently posed by the "millennium bug" may be traced to methodological laxity, and especially to inadequate documentation.

A methodology contains the following:

- Details of the activities that need to be performed during system development.
- Step-by-step instructions for each task.
- Techniques for use within the tasks (such as interviewing and modelling techniques).
- Documentation methods/formats.
- General advice and guidelines.

There are many benefits to be gained from using a methodology:

- Developers can benefit from the knowledge of experts in the field.
- Developers who are new to the field will not omit essential tasks.
- Standardised procedures mean that the work of an individual developer can be more easily followed by another.
- It may be possible to recruit staff already trained in a required methodology.
- Applications, or parts of applications, can be more easily adapted and reused.
- Ease of maintenance. The time and effort devoted to the maintenance of most applications is greater than that needed for the original development.

- Project management is greatly facilitated, as recognised stages and activities can be identified, and if necessary allocated to development team members.

4. Organisational size

The power of computer hardware and software is increasing dramatically, and prices are falling. KBE software is no exception, and is now within the reach of smaller budgets. However, the ability to meet these costs does not put SMEs on an equal footing with larger companies. Current KBE development techniques have evolved with the experience of large concerns, and may not be suitable for SMEs. This section examines some of the differences between large and small organisations that need to be taken into account when devising a methodology for SMEs.

- The lack of staff with experience of KBE systems, and IT systems in general, is a key factor. It is probable that the volume of work within these areas in an SME will not be sufficient to justify their employment. Consequently, the development of new systems may necessitate the involvement of an external organisation. The REFIT methodology aims to maximise the contribution of the client organisation so that the costs of employing external agents can be kept to a minimum.
- In times of difficulty, companies may shed staff in an attempt to cut costs. This can result in the loss of vital knowledge for any organisation. However, the danger is greater within an SME, where critical knowledge may be in the hands of a select few. It is obviously to the company's advantage to capture this knowledge for future use within a KBE system.
- The size of the team needed to develop an application will be related to the size of the application itself, and this is likely to be smaller for an SME than for a larger concern. Consequently, lines of communication between developers, as between employees of the organisation itself, may be less formal. This may well be advantageous in some respects, but a source of problems in others.
- The limited number of people with whom the developer can work within an SME may present difficulties. Confirming knowledge obtained from a single source is obviously problematic, leading to the possibility of errors, misunderstandings, omissions, or even deliberate misinformation.
- The organisation of a business may centre on functions or individuals. Consequently, when examining its activities, it may be appropriate to look at what is done by "the estimator", or by "Mr. Tate". Within a large company, the estimator could be expected to have a definable set of functions, all directly concerned with providing quotations in response to customer queries. In an SME, however, Mr. Tate might do the estimating, order materials on Fridays, and assist with any number of other tasks, as and when required. This will have a significant impact on how knowledge is acquired — who needs to be interviewed, and what questions should be asked.
- Finally, smaller companies may have more limited finances than their larger competitors. Whereas the latter may have reserves that can be drawn upon for investment in R&D in lean times, SMEs may not be in a position to engage in speculative projects.

5. Existing methodologies

Very little research has been published on methodologies for KBE system development. This may be a significant factor in the relatively slow uptake of the technology [5]. Furthermore, most of the existing literature focuses on KBSs, rather than KBE systems, and on the needs of large organisations.

Perhaps, the most widely known methodology for KBSs is CommonKADS. It supports project management, organisational analysis, knowledge acquisition, conceptual modelling, user interaction, system integration, and design. It describes KBS development both from a project management perspective and a results perspective. The latter views KBS development as the continuous improvement of a set of models of various aspects of the KBS and its environment [6]. In this respect, there are similarities between CommonKADS and the approach taken for the methodology that is the subject of this document. However, the individual models used are quite different. Perhaps more importantly, CommonKADS is large and complex, can be difficult to learn, and the effort required for its implementation would be disproportionate for small companies and small projects [7].

MOKA (methodology and tools oriented to knowledge-based engineering applications) [5] is a project that aims to produce a KBE system development methodology that will form the basis of a new international standard. It too relies heavily on industrial involvement (Coventry University is the only academic partner). It will be some time before the project is complete, and final results are published. Interim publications reveal similarities between the view of the lifecycle adopted by both projects. However, the needs of SMEs are highly specialised, and the differences will be reflected in the detailed contents of the methodologies.

Producers of software for building KBE applications naturally provide instructions for its use, and this sometimes extends to more general advice on project development issues [8]. However, this is highly product-specific, and unsuitable for the purposes described in this document.

6. Aims and approach

The principle aim of the project is to develop a methodology that is suitable for SMEs. Most importantly this means minimising the time, effort, and expense needed for system development, without sacrificing quality and maintainability. The approach to achieving this aim is essentially a practical one. The REFIT team is constructing demonstrator applications for, and in collaboration with, the following UK foundries:

- Bridge Foundry, Wednesbury, West Midlands;
- Armatage Shanks, Wolverhampton, West Midlands;
- PDC, Poole, Dorset;
- Butlers, Brownhills, West Midlands.

A demonstrator is an item of working KBE software, built to illustrate the potential of KBE solutions. It enables companies to evaluate the suitability of KBE for their application areas. An additional aim is to improve the profitability of the foundry sector by introducing skills that will be of use beyond the lifetime of the project.

The team also has the support of several UK trade bodies, including the following:

- Castings 2000,
- The British Investment Castings Association,
- The Institute of British Foundrymen.

The methodology will therefore be built upon the experience gained from demonstrator development and the associated industrial contact, as well as the usual more academic research activities.

7. The framework

It is a common practice to break down the activities involved in the development of an information system into groups, and to list them in the order in which they are carried out. This method is suitable for traditional systems in which the tasks to be carried out are fairly predictable. The nature of KBE systems is such that knowledge revealed in the early

Table 1
Activity groups and activities

Activity group	Activity	Deliverable(s)
Initial investigation	Strategic discussion	Business description, business objectives statement, level 0 activity diagram
	Introductory interview	Level 1 information flow diagram, rich picture of whole organisation
	Initial in-depth interview	Organisational structure chart, level 1 activity diagram
	Follow-up interviews	As necessary: lower-level activity diagrams, process diagrams, rich pictures
	Application identification	Proposal
	Cost/benefit analysis	Feasibility report
	Risk analysis	Feasibility report
	Project authorisation	Authorisation to proceed
Application classification	Application classification	Completed questionnaire or grid, application class ID
Requirements analysis	Functional requirements analysis	Function structure charts, function flow charts, function descriptions
	User interface requirements analysis	Screen layouts, textual descriptions, report layouts, interface prototypes
	Data requirements analysis	Data structure descriptions, conceptual database diagram
	Knowledge elicitation requirements analysis	Knowledge elicitation plan
	Performance requirements analysis	Performance requirements table
	Requirements analysis overview	Requirements specification overview
	Requirements analysis review	Authorisation to proceed
Tool selection	Hardware investigation	Hardware selection
	KBE software investigation	KBE software selection
Design	Functional design	Function structure charts, function flow charts
	User interface design	Screen layouts, textual descriptions, report layouts, interface prototypes, interface navigation diagrams
	Data design	Data model
	Knowledge capture	Rationalised knowledge model
	Testing strategy	Test plan and data
	Design overview	Design overview
	Design review	Authorisation to proceed
Implementation	Coding	Program code
	Code documentation	Program documentation
Testing	Integration testing	Integrated application
	Verification	Verified application
	Validation	Validated application
System realisation	Software installation	Installed system
	Training	Trained users
	User documentation	User manual
Maintenance	System maintenance	Maintained system

stages of development may heavily influence the tasks to be performed in the later stages. Consequently, there is a danger of trying to impose an artificial structure on processes which are inherently unstructured.

Activities within the methodology are divided into activity groups, and they are presented in an order that may approximate to that in which they will be carried out. However, the sequence is far from rigid. It is accepted that there will be many situations for which some of the activities will be inappropriate, and may be omitted. Iteration to previous activities, and the execution of activities in parallel, will also be extremely common. The methodology will provide guidelines on the correct choice and sequencing of activities, both for application classes and typical applications. An example of an application class is one that inputs numerical parameters and outputs geometrical figures. A typical application might be a product-costing program. The activity groups and the activities of which they are composed are shown in Table 1.

At this stage effort has been concentrated on identifying the activities to be performed, and on devising appropriate modelling techniques. In addition to analysing the needs of the organisation and designing solutions to the problems identified, there is a need to do so in an integrated and structured fashion, at the same time minimising the number of concepts and techniques employed.

8. The integrated modelling set

A KBE system can be regarded as a hybrid information system [9]. That is to say it has some characteristics both of KBS systems, and of more traditional information processing systems. Consequently, it is appropriate to make use of tools and techniques already in existence for both types of system. However, there are dangers in collecting fragments from a variety of sources, and simply throwing them together in the hope that they will form a methodology [10]. These include unnecessary duplication of tasks, inconsistency between methodology components, and missing or incomplete coverage of development activities. Fig. 1 shows the types of models that are to be employed within the methodology.

The modelling techniques selected for use within the initial investigation activity group will be the subject of a later paper. They focus on an activity-oriented view of the organisation, and include a version of IDEF0 diagrams [11] which has been adapted to add a knowledge-related bias. Rich pictures [12] are used to represent ‘‘soft’’ issues.

9. The methodology and SMEs

A previous section stressed the requirement for the methodology to meet the needs of smaller organisations. Having identified these needs, it is fitting to examine the ways in which they are being addressed:

- The disproportionate effect of a failed IT project on an SME will be reflected in an emphasis on risk analysis in the initial investigation.
- Activity models and the organisational structure chart are linked in recognition of the importance of individuals in an SME.
- The methodology stresses the importance of involving a wide range of company personnel — to encourage a sense

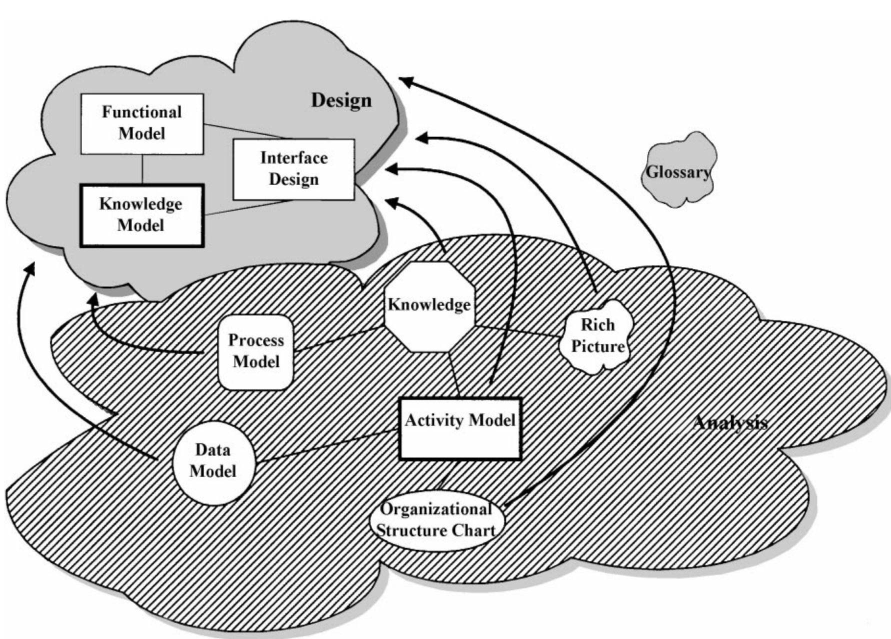

Fig. 1. The integrated modelling set.

of involvement in and/or ownership of the project, and to avoid a feeling of resentment on the part of employees.

- The informal lines of communication which often exist within an SME can mean that the management has an incomplete understanding of the ways in which information and knowledge are used within the organisation. Activity diagrams are used to clarify organisational structure, and to identify application areas suitable for KBE solutions.
- The methodology assumes little or no previous IT experience on the part of the users.
- The number of types of models is kept as low as possible to enhance simplicity.
- Comprehensiveness is maintained however, including documentation techniques and guidelines for all of the activities to help ensure that systems can be maintained easily and effectively.

To ensure that the methodology is workable, it will have a sound theoretical foundation, but will be heavily influenced by the practical experiences of the REFIT team in developing applications with their industrial collaborators.

10. Conclusions

The application of KBE techniques has provided large companies with powerful tools to enhance their business performance. Smaller organisations can also benefit from the discipline, but they lack the methodological guidance available to their larger competitors. The needs of SMEs have been evaluated and a methodology to support their particular requirements is under development. This will be strongly influenced by the KBE system development work being carried out in parallel with industrial collaborators to ensure the practicability of the finished product.

References

[1] C. Bancroft, Revitalisation of expertise in foundries using information technology, in: Proceedings of the 24th European Conference on Investment Casting, Rome, Italy, 1998, pp. 7.1–7.12.
[2] D. Calkins, Learning all about knowledge based engineering, Intelligent Manufacturing Report 2 (3) March (1996).
[3] C. Selbeck, Open architecture, Vertigo — The Official Newsletter of the International ICAD Users' Group, Vol. 7, No. 1, August 1998.
[4] M. MacLeod, See all, know all, tell all, Professional Eng. 11 (12) (1998) 19–20.
[5] K. Oldham, S. Kneebone, M. Callot, A. Murton, R. Brimble, MOKA — a methodology and tools oriented to knowledge-based applications, in: Proceedings of the Conference on Integration in Manufacturing, Göteborg, Sweden, 1998, pp. 198–207.
[6] R. De Hoog, R. Martil, B. Wielinga, R. Taylor, C. Bright, W. Van de Velde, The CommonKADS model set, ESPRIT Project P5248 (Document KADS-II/M1/DM1.1b/UvA/018/6.0/FINAL), 1994.
[7] C. Cadas, N. Parthenios, Catalyst — use of CommonKADS methodology in knowledge based system development, Report on ESSI Project 10327, 1996. WWW: http://www.esi.es/ESSI/Reports/All/10327/Report/index.htm on January 15, 1999.
[8] T.D. Stevenson, A project development strategy, in: Proceedings of the European ICAD User Group Meeting, Bristol, UK, 1998.
[9] X. Chen, S. Kendal, I. Potts, P. Smith, Towards an integrated method for hybrid information system development, IEE Proc.: Software Eng. 144 (5/6) (1996) 261–269.
[10] W. Wang, K. Popplewell, R. Bell, An integrated multi-view system description approach to approximate factory modelling, Int. J. Comput. Integrated Manuf. 6 (3) (1993) 165–174.
[11] Standard for integration definition for function modelling, Draft Federal Information Processing Standards Publication 183 (IDEF0), National Institute for Standards and Technology (NIST), 1993.
[12] P.B. Checkland, J. Scholes, Soft Systems Methodology in Action, Wiley, New York, 1991.