Two Case Studies in Expert System Development

Dr. Ian Watson
University of Salford
Bridgewater Building
Salford, M5 4WT,
UK.

Abstract

This papers describes two case studies of an expert system development methodology called the Client Centred Approach. The paper outlines the Client Centred Approach and contrasts it to KADS and other knowledge modelling methodologies. The paper then describes two case studies of the methodology in use in business. The first case study is of an unsuccessful development project and the second of a successful development project. The paper concludes by identifying some features that characterise successful expert system developments.

1. Introduction

Several years ago the majority of expert systems (ES) in business in the UK were developed by blue chip companies, such as ICI, Unilever, Shell and British Telecom. However, as a recent UK Department of Trade and Industry survey shows, smaller companies are increasingly taking up this new technology [DTI, 1992]. As ES have left the research labs more interest has been focused on methods to assist in building them, as it is now usually assumed that the technology is mature and stable.

Ironically, just as many developers in the traditional information systems community are moving away from prescriptive development methods such as SSADM towards more flexible contingent development methods (*e.g.*, Soft Systems and Multiview [Checkland, 1989; Avison & Wood-Harper, 1991]) the ES community is moving in the opposite direction (*e.g.*, KADS [Hickman et al., 1989; van Harmelon & Balder, 1992, Wielinga et al., 1992], the work of Chandrasekaran [1988] and Steels [1990]).

These techniques centre upon modelling activities without necessarily offering methods for implementing successful systems. However, the desire to have a single development method that will always be successful is a natural one. A step-by-step method provides security, particularly if one has never built or commissioned a similar system before. However, this desire is unrealistic, and dangerous. ES development methods must be contingent rather than prescriptive because the skills of different knowledge engineers and the situations in which they work must always be taken into account in any project. If this is now beginning to be accepted in the information system community then surely it must also be so when one is encoding human expertise and knowledge.

This paper presents an outline of a contingent ES development methodology called the Client Centred Approach. It is the result of three years of research funded by the UK Department of Trade and Industry [Watson et al., 1992a & b]. To put the methodology in its context the paper describes two case studies of the methodology in use. Following the maxim that you learn as much from failure as from success, the first case study is of an unsuccessful ES

Watson, I. (1994). Two Case Studies in Expert System Development. In, Proc. 2nd. World Congress on Expert Systems, Lisbon, Portugal, Jan 10-14, 1994.

development project. The second of a successful one. The paper concludes by offering some guidelines to good practice for developing ES.

2. Client Centred not Technology Centred

Until recently, ES technology was only available to large organisations. Therefore, the available development methods and tools available were designed to cater for their needs — this tends to emphasise the technology. The Client Centred Approach (CCA) places the clients at the centre of the entire development process and de-emphasises the role of the technology [Watson et al., 1992a & b].

The CCA enables practitioners to use the techniques and tools they are familiar with within a guiding framework. It is designed to help managers avoid wasting time and money developing systems that are not brought into regular productive use. The CCA pays particular attention to decisions made at the early stages of the system's development. This means the roles, benefits and limitations of the system must be clearly stated at the beginning of the project.

The CCA provides clearly defined stages or audit points to help managers guide the development process from the system's initial conception through to its eventual use within the organisation. Project development is controlled by a simple project management technique, based on Boehm's spiral [Boehm, 1986 & 1989], that identifies potential threats to the project's success [Bright et al., 1991; Watson 1992].

3. The Stakeholders

The CCA involves **all** the project stakeholders in the development process. A stakeholder refers to anyone who is or will be affected by use of the expert system. There are many of these:

- **The Client:** The person or organisation for whom the expert system is being built. The client owns the problem that motivates the expert system project and provides or authorises resources for the project.
- **Project Manager:** The person within the client's organisation responsible for the management of the project.
- **Knowledge Engineer(s):** An IT professional who obtains and organises the knowledge and programs the expert system. A project may employ several knowledge engineers from within their organisation or use consultants.
- **Expert:** Someone with expert knowledge in the problem area. There may be experts who provide knowledge and experts who check the system's results. The experts may be employed in the client's organisation or brought in from outside (e.g., academics).
- **Information/Data Supplier:** someone who supplies data or information that the expert uses to solve a problem. This person will need to supply the expert system with the same information.
- **Information/Data User:** someone who uses the solution made by the expert. The person may become a direct user of the expert system or they may obtain their information indirectly through a subordinate's use of the expert system.
- **Software & Hardware Support:** someone responsible for the maintenance of the client's software and hardware.
- **Customers & Suppliers:** the customers and suppliers of the client's organisation who may be affected by the installation and use of the expert system.

- **Anyone Else Affected:** The above list is **NOT** exhaustive. A stakeholder is **ANYONE** who will be affected by use of the expert system.

4. A Staged Method

The CCA is designed to assist organisations manage expert system development projects. It uses a seven-stage structure to guide the development from initial conception through to regular beneficial use of the expert system.

These stages are named after the deliverables the client receives throughout the project. Consequently, managers using the CCA (who are not expected to have backgrounds in IT) can make informed decisions throughout; that is, managers do not have to rely entirely on the advice of external consultants or in-house IT experts.

The CCA guides managers through the entire life-cycle of the expert system from initial conception to implementation and then to its eventual maintenance and updating.

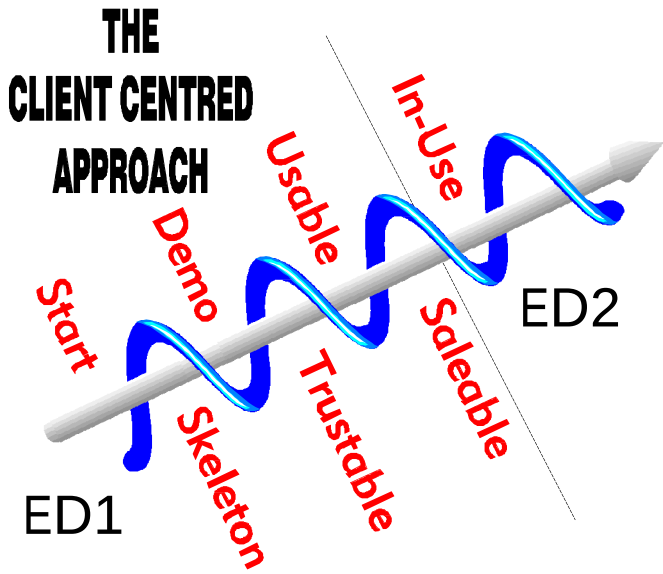

Fig 1: Seven Stages of the CCA

A central premise of the CCA is the recognition that an expert system is never finished and that knowledge continues to evolve after the system has come into use.

5. Life Cycle Method

The life-cycle method is based on several guiding principles:

- Spend time and effort at the early stages of an expert system project in defining roles, benefits, objectives and so forth;
- Keep a holistic approach throughout the project, include considerations of usability and saleability from the start;
- Functioning software is useful both as a discussion point, and for deriving a specification;
- The staged approach helps the clients plan resource provision;
- The iterative approach to knowledge acquisition and representation is necessary because of the well known problems of devising a specification in ill-structured areas;
- Involve the client and other stakeholders throughout the process;

- Speak the client's language — avoid IT and consultancy jargon.

5.1 Staged development

The project proceeds in stages so clients can plan resource provision. Because of the inherent uncertainty of expert system development detailed project plans can only be made from one development stage to the next.

The CCA is divided into two activities:

- **Evolutionary development part one (ED1).** This considers the development of the expert system until it is in regular beneficial use within your organisation.
- **Evolutionary development part two (ED2).** This considers how the system can be kept in regular beneficial use, and considers such factors as training of users and maintenance of the system.

ED1 consists of the following six stages:

5.2 Feasibility Study

The project feasibility study produces a total view of the project called a "Holistic Picture." This is used to guide the project.

The purposes of the first stage are as follows:

- to perform a feasibility study that includes business benefits, limitations and risk,
- to scope the expert system domain,
- to identify all stakeholders,
- to obtain a holistic picture of the problem by involving them all,
- to obtain their commitment to the project,
- to help the client plan resource provision,
- and to provide retrospective documentation of top-level decisions.

We recommend that developers should considering the following five questions (or hurdles) to focus their initial discussions:

1. Is the problem suitable for computerisation?
2. Is the problem suitable for expert systems?
3. Is the knowledge available to solve the problem?
4. Is the system worth developing?
5. Will the system be used?

An expert system is considered appropriate only if **ALL** the questions (or hurdles) are satisfied. Within this stage the development team are encouraged to use methods such as Soft Systems [Checkland 1989] or ETHICS [Mumford, 1986] to ensure that they have an understanding of the organisational context of the proposed system. However, conventional systems analysis techniques, such as context diagrams, data flows and entity relationship diagrams can also be used succussfully here, particularly if the organisation is already familiar with them.

5.3 Skeleton System

The Skeleton System shows a sequence of screens that give a look and feel of the overall functionality of the expert system. The Skeleton System contains little or no knowledge, but illustrates the overall functionality of the intended system. This ensures the clients and stakeholders understand what is intended. The Skeleton System also helps the knowledge engineers obtain an understanding of the domain and of the types of information handled. We have found that a skeleton system is extremely useful in obtaining resource commitment from senior management.

5.4 Demonstration Systems

Each problem has certain types of knowledge associated with it, and Demonstration Systems are designed to develop an explicit understanding of the types of knowledge and how they are interrelated. At this stage a significant amount of knowledge is acquired and modelled preferably in an intermediate knowledge representation. Techniques from KADS, Chandrasekaran's task analysis or Luc Steels' components of expertise may all be used if the developers have experience of them. Documenting the elicited knowledge is vital but it should also be encapsulated in a demonstration system that can be shown to the stakeholders (note: it is possible for there to be several demonstration systems each demonstrating different aspects of the system's functionality).

The Demonstration Systems keeps the stakeholders informed and involved. They provide the knowledge engineers with an understanding of the types of knowledge in the domain, and of the technical problems that will be encountered.

5.5 Trustable System

The final Demonstration System should give reasonable results in unexceptional conditions. However, the Trustable System gives correct results in all situations in which the expert system is expected to be used. At this stage validation and verification of the knowledge base occurs. This is easier if an intermediate knowledge representation has been used.

5.6 Usable System

The Trustable System gives correct results but will be difficult to use — probably only the people who built it can operate it. Therefore, it can not be used to bring real business benefit. This stage ensures that the expert system is usable, by considering features such as "what-iffing", explanation facilities, integration with other systems, and easy methods of data entry and export.

Note that giving too much effort prematurely to usability features is a common mistake, since developments can render them redundant. However, usability and such things as integration and documentation should be kept in mind throughout the project.

5.7 Saleable System

This stage prepares the expert system for wider distribution. Tasks here involve finallising manuals and installation routines, preparing training materials, establishing user support systems and preparing for the launch and subsequent activity. The deliverable is the final expert system and the business benefits it brings.

6. ED2: Embedded in Use

The seventh stage keeps the expert system in regular beneficial use, and represents the activities of ED2. This involves continuing to train users, providing help-lines and so forth, but it also means ensuring that any organisational changes required for effective use of the expert system are implemented. Crucially, the ongoing maintenance of the knowledge base will also have to be planned for and resources otained[Watson, 1992b].

The deliverables at the seventh stage are the business benefits that accrue from using the expert system.

7. Advantages of the CCA

Though it is a staged structure, the CCA overcomes the problems normally associated with linear structures because it allows for evolutionary development. Moreover, it also overcomes several problems normally associated with rapid prototyping [Berry & Broadbent, 1986]; namely:

- Systems may satisfy the expectations of the knowledge engineers and domain experts but not necessarily those of the users.
- Prototyping can lead to uncontrolled growth of the system causing difficulties in maintenance.
- Over reliance on the iterative refinement of expert systems can result in their inability to solve problems occurring only rarely in the domain.
- The final “delivery” system is often a tidied up version of the prototype that was being worked upon when either the project's time or money ran out.
- Because of its cyclical nature prototyping is very difficult to audit.

However, although others recognise that “people” should be at the centre of methods, they do not identify the cause of the above problems, namely that they are derived from being technology centred [Diaper, 1989]. Both linear and rapid prototyping methodologies are described in terms dictated by the technology (e.g., elicit, represent, validate, debug), and many omit to establish an initial holistic view. Thus, the client is not fully involved and the above problems result.

The criticism of being technology centred also may be levelled at KADS and other knowledge modelling methodologies [Chandrasekaran, 1989; Steels 1990]. They are full of jargon and describe activities in terms of layers of knowledge and ontological models. Such an approach will inevitably stakeholders clients with little or no experience (or interest) in ontology [Church 1993].

8. An Unsuccessful Case Study

Harris Lawson and Everard (HL&E), a large UK property consultancy company decided to develop an ES to advise on construction procurement methods. A senior partner of the company was due to retire shortly and the company thought it would be a good idea if they could retain his life times experience as a corporate asset. The company used an ES in its daily work and approached the developers of this system for advice. The developers of this system had used an early version of the CCA and advised HL&E to use the CCA. HL&E contacted the author and received a draft copy of the CCA manual and some training. The author was also available as a consultant to advise during development.

HL&E were prepared to invest six months effort of one of their IT specialists to develop a Demonstration System. They would then assess it and decide whether to invest further. The person chosen to develop the system was the manager of the company network. He had good IT knowledge, although no formal IT qualifications, and was a reasonable programmer particularly in spreadsheet development. He had no previous experience of ES development or of knowledge engineering techniques.

Although they were in possession of the CCA manual and had received training, during HL&E's feasibility study they did not adhere to several tenants of the CCA.

- They did not involve all stakeholders in the development, instead only three people in the company really new about the project: the developer, his manager, and the retiring expert.
- They did not scope the project down to firm boundaries, instead the system was to be all things to all people, providing consultancy advice, being used for training, as a checklist, and even for marketing the company.
- They did not clearly identify the business benefits, and importantly the limitations of the system.
- Finally, they did not allocate reasonable resources only six months and no real budget.

At the end of their feasibility study HL&E had only the vaguest idea of what the proposed system would do, but they decided to continue assuming that the function of the system would become apparent once programming produced some results. HL&E did not produce a skeleton system and show it to stakeholders. Instead they commenced knowledge elicitation.

Although they were advised to model the knowledge on paper first, using decision trees, decision tables, or inference nets the developer and the retiring expert were keen to "see how the knowledge would work". No decision had been made at this stage on what software to use. However, the author had developed a tool that lets decision trees be created on screen and immediately turned into an executable system [Watson, 1993]. The developer and the expert liked this tool since it gave instant feedback but they were advised that decision trees would almost certainly prove to be a restrictive knowledge representation and unsuitable for their task.

Nonetheless they decided to use the decision tree tool and take it as far as they could. Knowledge elicitation continued for several weeks with the decision tree tool being used to represent the knowledge. Inevitably, they quickly ran into the limitations of decision trees, even ones that were not binary and could be formed into complex graphs. The expert frequently provided information as anecdotes of particular projects and was often unable to generalise rules since the outcome of different projects was contradictory. HL&E were advised to consider case-based reasoning but they felt this technology was not for them as they wanted a "definitive answer".

It became apparent that HL&E had a naive view of ES technology. They assumed that because expert systems were new and were a product of Artificial Intelligence research that the software would be "intelligent". They were surprised that ES had to be "programmed" in a relatively conventional manner and that powerful development environments such as Kappa, Art or Nexpert would take several months to master. Because HL&E's developer still had to mange the company network he was unwilling to learn to use a tool such as Kappa. Moreover the company were unwilling to invest several thousand pounds in purchasing such software.

Knowledge elicitation continued for several more weeks with knowledge being recorded in taped interviews. The developer then became ill and when he came back to work six months were up.

It was not a difficult decision for HL&E to make. They had not produced a credible demonstration system in six months (even though the developer had not really been full-time on the project). They were disillusioned with ES software, it seemed either affordable and easy to use but restrictive, or it was powerful, expensive and hard to learn. They were not prepared to make the investment in acquiring ES development skills, and consequently the plug was pulled on the project.

9. A Successful Case Study

Law & Mutual (L&M) are a major UK provider of financial services. Their IT department has an annual budget of around £60 million. The company was in the process of down-sizing from dumb terminals attached to mainframes to PC based LANs. As part of a business process re-engineering project they wanted to provide technical assistance to employees purchasing PC's, peripherals, software or upgrades. The author was asked to join the feasibility study and persuaded them to use the CCA as a framework within which to develop the system.

L&M put together a reengineering team that comprised a project manager for the entire project, a project manager for the implementation, a consultant systems analyst, a consultant knowledge engineer, several programmers, and the intended managers and users of the reengineered process. A week was spent in April analysing the existing process and reengineering it. This resulted in a very much simplified design whereby all L&M's employees would have access to a single point of contact for ordering PC products and upgrades.

At this time a feasibility study was produced that encompassed most of the elements required by the CCA along with a detailed cost/benefits analysis. A skeleton system was then produced using Asymetrix Toolbook and demonstrated to senior management within L&M. Management then decided to proceed with the project.

An essential component of this system was an expert system that would contain knowledge about L&M's IT strategy, their approved product range and the hardware/software that different business units used. Through an integrated database the ES could obtain the business unit and location of an employee and details of the network server they either were connected to or should connect to. Because of the volatile nature of the PC market it was decided that it would be important to make the maintenance of the ES as easy as possible. Case-based reasoning was chosen as the knowledge representation paradigm that would meet these constraints [Vargas & Raj, 1993]. Inference's CBR-Express and Microsoft's Access were chosen to develop the demonstration system. After approximately one months work a demonstration system was shown at several seminars to stakeholders from all the business units within L&M.

These presentations were carefully organised as part of a comprehensive communications plan. They were professionally conducted and involved describing why the existing process had to be reengineered, what benefits the new process would deliver, and concluded with a demonstration of the software supporting the process. These sessions were essential in obtaining the support of the whole of the company to the project.

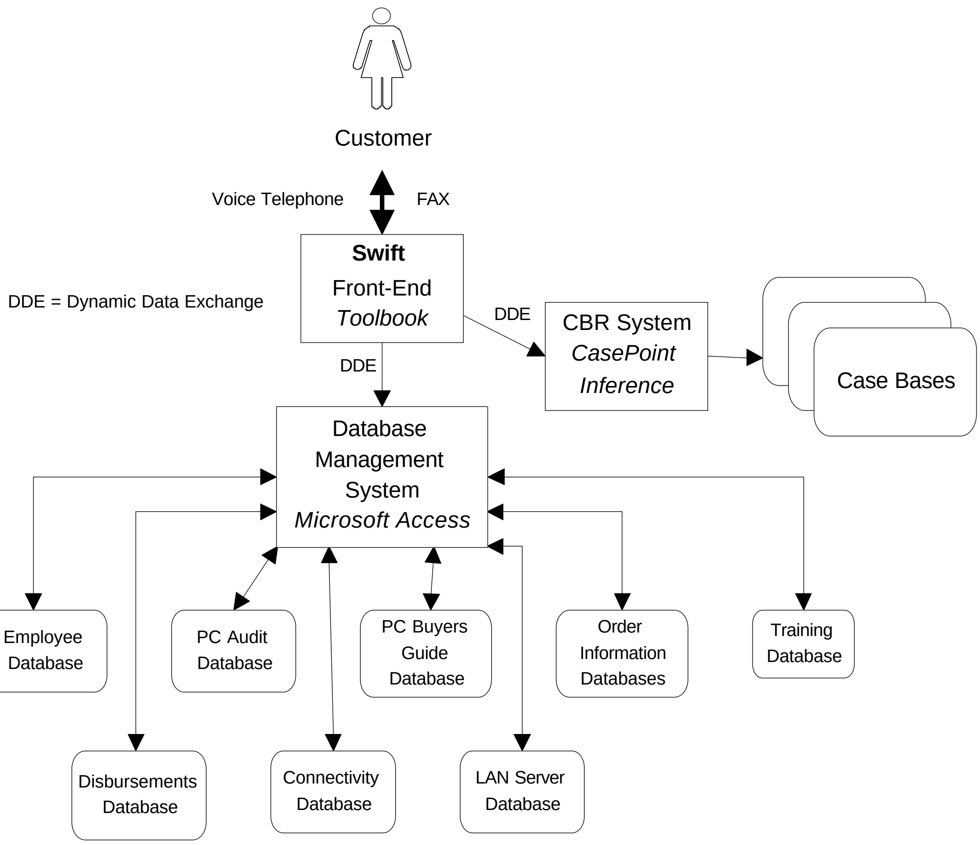

Fig 2: L&M System Architecture

Following this an intensive system development stage took place that effectively combined the Trustable and Usable stages of the CCA into one phase. This was partly because using tools like Asymetrix Toolbook, with its powerful GUI builder, makes prototyping the interface that much easier. During the final weeks of the development, the team that would man the service were employed. They were involved in commissioning the system. This was particularly valuable in prototyping the interface since the developers could make an alteration and obtain immediate feedback.

At the beginning of September, less than five months after the project started, the system went live. L&M had decided to roll the system out incrementally:

- first it would be used by selected people on selected projects that would be carefully monitored enabling bugs, technical difficulties, organisational and communication problems and simple oversights to be solved;
- then two weeks later it would be rolled out to serve the whole MIS department; and finally
- it would be rolled out business unit by business unit over six months.

This phased process was crucial since it allowed problems to be trapped early and avoided many of the problems associated with a *big bang* approach to delivering a new system. By Christmas of that year the system was nearly operational across the entire company, almost seven thousand employees, and was judged a success.

Finally, L&M had a made a commitment to maintaining the system. Regular monthly management meetings are held at which problems with the software are identified along with changes to the case-bases. recognising that knowledge changes is particularly important in the domain of personal computing where new products are brought out almost daily.

10. Conclusions

The CCA is exploring our understanding of the development of expert systems. We believe that there are two underlying problems with many expert system development methods:

- they are "technology centred," focusing on modelling techniques, and
- they are attempting to be prescriptive or exclusive.

The CCA has been shown to work on several development projects. Our recent experiences have enabled us to identify common features that successful expert system developments have had.

1. Involve all the stakeholders from the outset.
2. Spend time (and money) on a detailed feasibility study.
3. Identify the benefits the system will bring and identify how the benefits will be measured.
4. Plan to implement the smallest useful system possible, rather than a complex system to solve all your problems.
5. Develop a communications plan to sell the idea and the reality of the system to your organisation.
6. Manage the implementation closely.
7. Employ experienced developers.
8. Plan a phased system roll-out
9. Plan to maintain the system and obtain resources for this.
10. Above all be committed to the project.

Of the above features of successful developments, perhaps the last point is the most essential. The L&M project, for example, was characterised by the extraordinary level of commitment that everyone had to the project. This included the development team but more importantly senior management who were totally committed to the project's success and ensured that the development team were supported at all times.

We shall continue to develop expert systems and to develop the CCA. The objective of these studies is to gain a better understanding of the needs of developers and of the realities of developing expert systems. Not all of our efforts will be successful, but we find that we can learn as much from failure as from success. We are committed to developing a pragmatic methodology that combines the strengths of the software engineering linear approach (i.e., the waterfall) with the iterative approach of AI while overcoming many of their weaknesses.

11. Acknowledgements

I would like to acknowledge Andrew Basden who was responsible for originating the ideas behind the CCA, and Peter Brandon who obtained research funds to develop the ideas. I would also like to thank all those involved in developing the systems described in this paper.

12. References

Avison, D.E., & Wood-Harper, A.T. (1991). Information Systems Development Research: An Exploration of Ideas in Practice. *The Computer Journal*, Vol.34(ii): pp.98-112.

Watson, I. (1994). Two Case Studies in Expert System Development. In, Proc. 2nd. World Congress on Expert Systems, Lisbon, Portugal, Jan 10-14, 1994.

Berry, D.C., & Broadbent, D.E. (1986). Expert Systems and the Man-Machine Interface, in *Expert Systems*, Vol.3(iv): pp.228-231.

Boehm, B. (1986). A spiral model for software development and enhancement. *IEEE Computer*, May 1988.

Boehm, B. (1989). *Software Risk Management*. IEEE Computer Society Press.

Bright, C., Martil, R., Williams, D., & Rajan, T. (1991). The Kads-II Framework for ES Project Management. Proc. *Ist SGKBS Int. Workshop on Knowledge-Based Systems Methodologies*.

Chandrasekaran, B. (1988). Generic tasks as building locks for knowledge based systems: The diagnosis and routine design examples. *The Knowledge Engineering Review*, 3(iii): pp.183-210.

Checkland, P. (1989). Soft Systems Methodology. *Human Systems Management*, 8: pp.273-89.

Church, C.. (1993). A Business Plan for Knowledge-Inclusive Systems. In, *Research & Development in Expert Systems X*, Bramer, M. & Macintosh, A.L. (Eds.), pp.3-16.

Diaper, D. (1989). *Knowledge Elicitation: Principles, Techniques & Applications*. Ellis Horwood Ltd.

DTI, (1992). *Knowledge Based Systems Survey of UK Applications*, Department of Trade & Industry.

Hickman, F., Killin, J., Land, L. et al. (1989*). Analysis for knowledge-based systems: A practical introduction to the KADS methodology*. Ellis Horwood, Chichester, UK.

van Harmelen, F., & Balder, J.R. (1992). $(ML)^2$: a formal language for KADS models of expertise. *Knowledge Acquisition*, 4(i). Special issue on "The KADS approach to knowledge engineering".

Mumford, E. (1986). *Using Computers for Business Success: The ETHICS Method*. Manchester Business School.

Steels, L. (1990). Components of Expertise. *AI Magazine*. Also available as: AI Memo 88-16, AI Lab, Free University of Brussels.

Vargas, J.E., & Raj, S. (1993). Developing maintainable expert systems using case-based reasoning. In, Expert Systems, 10(iv): pp.219-26.

Watson, I.D. (1992).
Project Management for the Evolutionary Development of Expert Systems. . In: *Research and Development in Expert Systems IX*, Bramer, M.A., Milne, R.W., (Eds.), pp.61-73. Publ. Cambridge University Press, Cambridge, UK. ISBN: 0 521 445175

Watson, I.D., Basden, A., Brandon, P.S. (1992a). The Client Centred Approach: Expert System Development. *Expert Systems*, 9(iv): pp.189-95..

Watson, I.D., Basden, A., Brandon, P.S. (1992a). The Client Centred Approach: Expert System Maintenance. *Expert Systems*, 9(iv): pp.181-88.

Watson, I.D. (1993). The Decision Tree Maker: An Educational Tool for the Representation of Knowledge. In, *Proc. AI in Education (AI-ED 93),* Edinburgh, August 1993.

Wielinga, B., Van de Velde, W., Schrieber, G. & Akkermans, H. (1992). The KADS Knowledge Modelling Aproach. In, *Proc. of the Japanese Knowledge Acquisition Workshop* (JKAW'92).

Details of where this paper has been sent:

Applied Expert Systems as *2 case studies in ES development*

Applied Intelligence as *Case studies in ES development*

Applied Artificial Intelligence as Expert Systems Development: 2 Case studies

Knowledge-Based Systems as The application of an expert system development methodology

Expert Systems as The Client Centred Approach: 2 case studies

The Journal of Information Technology as Developing Expert Systems: 2 case studies

The Computer Journal as An Expert System Development Methodologies: 2 Case Studies