*The Knowledge Engineering Review*

# Expert systems in UK business: a critical assessment

A. D'AGAPEYEFF AND C.J.B. HAWKINS
*Consultants in Information Technology, 60 Wildwood Road, London NW11 6UP, UK*

## 1 Preamble: observations and acknowledgements

The main purpose of this survey was to update a survey report prepared for the UK Alvey Directorate during 1984. The update was focused on business, and had to be qualitative because the increased confidentiality associated with the majority of projects precluded a quantitative analysis, and is confined to a synthesis of the impressions we have obtained. The survey was not intended to be representative. On the contrary, organizations were identified as being amongst those most likely to provide interesting application experience. This led to an arbitrary selection. For example, all banks were excluded because it was felt they were not yet ready to talk of their experiences. The companies approached consisted of five organizations visited in 1983, five further user companies and five service companies (e.g. consultants and software houses).

With one exception we were able to visit all those on the original list. The persons met are shown in Appendix A. They have been able to comment on an earlier draft of this report but the spectrum of opinions precludes any claim that it has universal agreement. Yet we do believe the great majority would support the principal findings and conclusions. We asked those who had relevant applications to submit a description of one project that might be useful to likely readers of this report. Some participants were not allowed to give out any project descriptions, but those that were submitted are shown in Appendix B.

In what follows we shall frequently refer to the 1984 report as a base point for comparison, and perhaps surprisingly, much of it still appears valid. For example, most operational applications remain "Simpler Expert Systems", as was then described, but with noted extensions to their purpose and to their component expertise.

### *1.a Principal findings*

(a) There has been a substantial growth in the number of project teams since 1984, most large organizations now have some such team, but experience of operational applications is still limited and narrowly held.
(b) The pinnacle of company experience appears to be 8–12 operational applications, but such teams are rare and, apart from suppliers, are found only in giant corporations.
(c) A second, and slightly larger, band of experience consists of 2–5 operational applications, chiefly produced by tiny teams starting with modest goals, while some famous names in the field have yet to deliver a system into business usage.
(d) Nevertheless it is believed that the pace of development is accelerating and the count of operational applications should, on a national basis, rise sharply within a year.
(e) At the same time the quality of exploitation, although improving, continues to be constrained by the lack of management commitment, by business secrecy, by poor organization and by a fear of the nature and the costs of this technology.
(f) The fear of high costs for expert systems has been corrected recently by Tim Johnson of

This report is based on a survey originally prepared under the aegis of the UK's Alvey Directorate.

Ovum: "our studies show that work stations and their associated American software have made very little impact on British business applications which is a market dominated by British personal computer shells"—the present study supports this observation.

### *1.b Conclusions*

Existing operational projects are relatively simple in form and scope. This could be a strength in correcting the widespread mis-conceptions about expert systems among executives. However they should be treated more vigorously as a stepping stone toward the tackling of really significant applications. For example, more of them could be assisting the staff by the distribution of critical expertise, assisting line managers by contributing to their decisions-making and assisting the service to customers by increasing the flexibility of software for front-office systems. If expert systems, on their own or otherwise, can be expected to have a significant impact on the competitive ability of companies, then senior management in Britain must:

(a) enlarge their own understanding and involvement in the implications of this technology;
(b) determine their on-going application priorities within an overall strategy for IT;
(c) provide an organization that will reflect those priorities and ensure the steady accumulation of project and technical experience;
(d) reduce to a minimum the adverse effects of project secrecy.

There is nothing in expert systems which confines them to large organizations except a misguided view of their purpose and nature. Every company beyond the smallest should, at least, consider their adoption.

Finally, any suggestion that the aims of current business applications either could, or should, reflect the special qualities of human beings, such as learning or intelligence, is totally mistaken.

### *1.c Amplification of findings*

There are clearly marked distinctions between:

(a) the perception of expert systems (e.g. uniquely clever programs having a wide scope);
(b) the goals of researchers (e.g. targetted at very long term aims);
(c) the limited benefits of existing applications (e.g. an unexpected usefulness in very narrow domains with no innate cleverness).

but with little understanding as to why they are so different.

A mixture of incompatible views is held by, and puzzles, business personnel. One consultant said "Managers in our clients are just beginning to get over their fright at expert systems". It is essential that they do so because their choice of applications must stem from a cool and sound appraisal of current needs and achievable goals. The latter will vary with advances in the technology and the experience of the local implementation teams.

Little evidence was detected of senior management direction. Partly this was a result of a self-imposed handicap. The great interest of management is the profitable and competitive advantage obtained from these systems. They do exist but the nearer a project comes to achieving them the more restrictive is the confidentiality imposed on the project details. Hence it is too easily believed that "we are ahead of anyone in our industry", which may be untrue for (say) overseas competitors. Projects may also progress without any real urgency, be overly dependent on a single departmental manager, and fail to be exposed to critical review, especially as to their knowledge engineering.

The study itself was skewed because those visited were more able to talk about their simplest applications. Nevertheless the impression obtained was that the more experienced teams were capable of expansion, could progress faster and could undertake more ambitious applications. In our opinion these teams might also gain from a closer co-operation with research groups having a practical interest in the applied side of expert systems. It is of interest here that a few individuals in companies and universities would like to review and strengthen the pattern of co-operation already initiated under the aegis of the UK's Alvey Directorate.

The level of investment required in most installed applications remains relatively low. There are occasional projects with large teams, one numbered 15, but the typical range of effort is 3 man months to 2 man years prior to maintenance. This, the care and updating of the knowledge bases, is a serious responsibility, but it is far less onerous and more immediately rewarding than for large COBOL programs. One unconfirmed but apparent finding could be a cause for concern. Large applications are not merely smaller projects that have grown. They have to be so designed. On the other hand, starting with a large project does appear to have a greater risk of failure, not necessarily from any particular fault in the program, but primarily due to subjective human reactions.

Yet it is entirely inappropriate to end these findings on a dismal note. Practitioners do differ in their views (e.g., on the quantitative treatment of uncertainty in the programmed inferencing) but, collectively, they are tremendously optimistic about the future potential of their craft. This optimism seems justified. They have seen users eyeing the screen in wonder as expertise, not previously put into writing, is presented and made pertinent to the contexts which are important to those users. Applications are not inevitably a great success but many do stimulate an enthusiasm among users that is rare in computing.

It is very desirable that these benefits are spread more widely. Almost every company has a myriad of applications simple enough to be built cheaply, albeit with caution, by comparatively novice knowledge engineers. This is how many of the giant corporations actually started. Provided that the slowly accumulated skills of building these applications are nurtured and husbanded, the applications can be gradually extended and integrated with relevant procedural software. At the other end of the scale, a few experienced teams have reached the stage of contributing, for example, to sophisticated decision support systems for which British industry has a vast need.

## 2 The nature of these systems

### *2.a The confusion*

One reason why some senior managers hesitate to concern themselves closely with the exploitation of expert systems is the contradictory explanations they receive as to what these systems are, or should be. At least three categories of opinion are on offer from competent, middle ranking, executives. These are:

(a) the whole area is an expensive dream which could cause an unnecessary diversion of specialist staff;
(b) the systems are a novel method of programming, chiefly for the sharing of expertise and the enhancement of conventional software;
(c) The systems are a revolutionary approach to computing which borrows from, and potentially improves upon, human capabilities.

In these circumstances it is understandable for management to pause until the matter is resolved by the emergence of a consensus view. Unfortunately there are too many conflicting interests for this to be an immediate prospect. Sadder still, it might already have been accomplished if there had been a greater readiness to listen to practitioners who had already built operational applications and if those practitioners were able to speak more convincingly of the benefits achieved. But the latter could not be sufficiently convincing and, after all the furore about the subject, few commentators wanted to highlight simple starting applications. Giant oaks may grow from modest acorns, but that is not the stuff of headlines.

### *2.b A revolutionary advance?*

The concepts derived and explored by AI scientists are fascinating, have broad ramifications and herald a profound, if eventual, change to software. Western research was openly acknowledged as a major source of the plans for the Japanese Fifth Generation Computer Project which was deliberately mounted as a revolutionary strategy for computing. These concepts and those plans are enriched by anthropomorphic terms (e.g. learning) which have a direct meaning for ordinary people about themselves. Therein lies the problem.

The terms used by researchers, in general discussion, should not necessarily be taken literally or as a forecast of early applications. The aftermath of the Japanese announcements were very exciting times for those in expert systems. Suddenly the area was topical and nationally important. AI Scientists whose studies had hitherto been designated as a remote backwater, were feted and funded. Their views were sought on a number of topics. These are circumstances in which the judicious choice of words is difficult. Scientists are not the only people who, convinced of the importance of their goals, can be inclined to believe they are also imminent. In speaking both directly and indirectly to businessmen, then and subsequently, the contrast between the "do-able now" and "ultimate aims" was not always well drawn. The media, unaware of the size of the gulf between the two, sometimes muddled them together. In any event the interests and aspirations of AI science became predominant in the limited public discussion on expert systems.

Revolutions, in the sense of sharp discontinuities to previously supposed foundation assumptions, are somewhat popular in research. In business they are normally symptoms of a disaster. Managers, unless compelled by events, will be disinclined to accept radical changes to existing methods without being persuaded of the advantages which flow from those changes. They may or may not be aware that their simpler applications are firmly rooted in the work of Feigenbaum and others at Stanford University, but the evidence of an imminent revolution in computing is unproven. In our opinion it is likely to remain so for some years.

### *2.c The current reality in business*

Phrases like "artificial intelligence", "machine learning" and "superior reasoning systems", relative to a human context, have no meaning whatsoever in current applications. There is nothing remotely akin to natural intelligence or to human reasoning that is being, or could be, programmed now in business. Assertions to the contrary, from any source, can damage the willingness of people to participate in sensible, if initially modest, projects. In the case of "learning" there is a tincture of apparent justification from the capacity of some software to generate valid rules by induction from well chosen example data. Induction may be on course to become a routine tool in the acquisition of expertise. However, the program is entirely unaware of what exactly it has "learnt" and the generated rules should always be subject to human checking.

The other main anthropomorphic term is knowledge. Most study participants would say they build knowledge based systems (KBS) and would claim that the knowledge was of human origin. KBS is a popular title and that may be deemed sufficient to override minor imperfections of meaning (but the same applies to the label AI). However, there are objections to the term knowledge. The term implies cognitive processes or brain content but applications are, at best, built only from its external manifestations. These may be in the form of rules articulated by the experts or through interviews about their skills. There is also the need to encompass within the system the fashion, in some quarters, of modelling aspects of the domain either numerically or by qualitative relationships. Thus, in this paper, "expertise" has been used as the most generic description of knowledge base content.

Applications exploiting expertise, preferably in an explicit form, are in truth an evolutionary development for industry. The comparative simplicity of many current projects provides the opportunity for the new approach to be studied, refined and its future import to be assessed. There are, at least, two big lessons. First is the value being placed on selected personal but unsubstantiated experience. In certain circumstances a program element may reflect solely what Fred says he does, if Fred is the expert and other specialists find his rules plausible. Second is the escalating importance of end-user reactions in the building of successful applications. If the objective is the transfer of expertise to specific users then success means doing whatever it takes to enable those users to gain that expertise. The extent to which orthodox methods were used in the program building is secondary.

Gradually, as abilities in developing practical applications accrete, the fetters of simplicity can be slowly relaxed. In some degree simpler systems only means the projects that a team feel confident of implementing successfully. The responsibility for distributing any application to hundreds of remote sites is far greater than releasing it to close colleagues of the expert. Distributions to 100+ sites have taken place in, for example, the oil industry. Another goal can be a greater understanding of how certain decisions are actually taken in practice as opposed to the pretence of logical steps employed in justifying those decisions. Applications can also be combined to provide the user with a more valuable resource. Counts of the number of rules can be misleading, since the shells differ in this respect, but the size of projects is definitely increasing. Furthermore, the scope of applications is growing, both in the expertise being shared, and in the software being enhanced.

Nothing in the above is to deny the value of research. The automation of fully intelligent behaviour may appear to be a somewhat improbable attainment but it is a powerful stimulus to patient investigation. AI scientists discovered, long ago, that empirical expertise could be programmed to tackle problems that were otherwise intractable. They have also, inter alia, produced a representation that achieves the criteria of modular programming and can often be made intelligible to non-programmers. It is important to business that pure research in this field is adequately supported because it is impossible to forecast what has still to be discovered.

## 3 Systems descriptions, expertise and purpose

### *3.a Systems descriptions*

It remains true, in the words of the 1984 Report, that "there is no commonly accepted definition of expert systems". Indeed, a standard definition is now rather a forlorn hope. The detailed descriptions of that report retain a value for newcomers to the field. However, the main characteristics outlined then must be redrawn to reflect a wider range of content. There was a period when "an explicit representation of empirical knowledge" was what determined that a program constituted an expert system. It seems that the majority of practitioners still hold this view. For them the intelligibility of the application, including its inferencing, has become even more important for the experts in providing and checking their input. One piece of advice was "never let the knowledge engineers think their job is to improve the expertise as opposed to its presentation".

In fact, even during the 1970s when Stanford University was doing so much to create the technology, there was a different perception of the subject pursued by those working in logic programming. More recently, the techniques and tools of expert systems have reached the hands of other disciplines whose members may, or may not, call their programs by that title. One resulting variant is domain modelling, such as the mathematical representation of the functioning of some entity (e.g. a circuit) within an application. This may be chosen as relevant expertise when there is no available expert experience (e.g. the operation of a newly designed control unit). There is a general preference to combine these models with whatever expert rules-of-thumb can be tapped in a particular case. However, a few teams do seem to favour these models because they are in some sense "proper knowledge", as against the vagaries and uncertainties of empirical expertise.

Another variant may be regarded as the DP perspective. This is characterized by a concern with control, with integrating the expertise alongside existing software and with central responsibilities for standards, maintenance and training. The perspective can also be associated with some distrust of "users doing their own thing". DP interests do include tasks for which there is no external expert (e.g. enhancements to an operating system or to retrieval facilities from a database). They may have no desire for the end-users to find the inferencing intelligible and there are situations where this would be hard or impossible. Hence the systems are very close to normal programming.

Do expert systems, therefore, mean whatever anyone chooses they should mean? Evidently a move in that direction does exist. But we think a differentiation remains. It is in the style of declarative rather than procedural programming. The expertise is normally explicit and often empirical. Nevertheless, there are DP departments to whom all programming belongs to a single discipline. They may also feel that big applications should be built under their leadership and remain unconvinced that knowledge engineering has special aspects. We are in no doubt that it does have special aspects but we would not question the large influence that DP Departments will have on the future of expert systems in business.

### *3.b Expertise and its gathering*

The empirical expertise required for an application is describable as the rules-of-thumb confidently held by a reasonably articulate authority in the domain. There can, on occasion, be doubts as to who is the authority but, in the absence of convincing evidence to the contrary, it is the person who actually does the job and is regarded as the expert by his or her peers. The caution over convincing evidence arises because management can propose the theoretical rather than the real expert (e.g. the specialist in Head Office who does not have rules-of-thumb from recent job experience). Peer opinion can be a better guide although the Head Office specialist may act as an excellent reviewer of the knowledge bases once written. Management can also be surprised when they discover how the job is done, today, in practice.

Empirical expertise is gathered in three main ways of which some may be combined in a given application. They are:

(a) the expertise is programmed by the expert, or experts, to a greater or lesser extent, by themselves in the language of the shell product;
(b) the expertise is programmed by the knowledge engineer working in tandem with the experts (i.e. the programmed inferencing is intelligible to the experts);
(c) the expertise is separately elicited and programmed by the knowledge engineer from interviews with the experts who also participate in the checking, but more distantly than in (b) (i.e. the inferencing is not fully intelligible to the experts).

Among the above (b) appears to be the most popular and may remain so although some practitioners warn that intelligibility will grow more difficult as applications expand in size and scope. It might be assumed that almost the whole of the knowledge bases are composed of rules-of-thumb, but this may be far from true. For example, once there is an interface to external files, additional information can be made available which is considered "valuable for the users". Eventually the proportion of empirical expertise may become quite small but remain crucial.

Non-empirical expertise is summarized less easily. Our impression is that the implementors continue to feel their way. Aside from different forms of modelling, programmers have a greater freedom in making functions declarative. This may begin as a small departure from decision, or control, tables. Such table entries can be read as restrictive rules (e.g. when $a = x$ and $b = y$ then do routine number 23). The rules may be made more explicit and extended by the option of varying lists of clauses in both the condition and the action parts. There may also be a greater reliance on qualitative values, on inferencing from general symbol manipulation and by alternative representations. The facilities may be used, say, to increase the flexibility of an old program, to add a touch of expertise about particular users and to explore the advantages of the system having a snippet of self-knowledge. Secrecy prevents a measure of how far this work has progressed or what elements of application expertise have been included. This does not signify any lack of value, rather the reverse.

### *3.c Purpose*

The distribution or sharing of empirical expertise among various users has a diverse spectrum of purposes which include the following:

(a) to capture rare or disappearing expertise (also called rare skills archiving) but the ultimate users may not be identifiable, which can limit the tailored structuring and finishing of the knowledge bases;

(b) to provide a tutorial in some domain (i.e. the transfer of the expertise is of the essence);
(c) to allow partial experts and others in the domain to substitute for the expert (who may not be present in, say, overseas sites) but the recognition of the limits of the system may be crucial;
(d) to assist toward improvements in human performance as to methods, procedures or judgements (e.g. prompting the consistent use of appropriate methods);
(e) to explain a manual or reference document typically, but not universally, as to its interpretation in practical contexts;
(f) to aid the work of experts and their colleagues.

Non-empirical expertise is again difficult. The purpose is likely to depend on whether it is a new program in itself (e.g. a process control application designed to exploit expert systems), a fragment overtly added to an existing application (e.g. a plant scheduling system being enhanced by the expertise) or a piece embedded within some system software (e.g. a control program). There are a few of the earlier computing applications being totally recast to take advantage of this technology and proportionately more that have some extension grafted on them. Professional programming teams embedding expertise were reluctant to talk even in generic terms. Some of them seemed so antagonistic to the mystique of expert systems that they saw no advantage in making their customers aware of what they were doing. A common remark was "it is only another software technique".

## 4 Project teams, organization, training and tools

### *4.a Project teams*

The range in size of project teams is quite broad. Yet the average team numbered only two or three. Even that number can be misleading because it might include the expert for this particular project or a programmer on assignment. Hence, especially in decentralized organizations, there were often only one or two knowledge engineers at a given site. This is enough to make a start and to implement small applications, sequentially, but it is insufficient to build a professional cadre.

Several disadvantages emerge from these small teams. They are obviously sensitive to a person being ill or leaving. The view of its members are not adequately challenged or expanded by exchanges of project experience. The applications become shaped by team prejudices. It may well be that the quality of many business applications is more dependent on their knowledge engineers than their experts. This apparent heresy occurs because deficiencies in their component expertise can normally be rectified subsequently but the user departments will regard weaknesses in their knowledge engineering as inherent to the technology. Hence the importance of a sound organization for the projects.

### *4.b Organization of knowledge engineering*

There is no single correct plan by which knowledge engineering is organized. A particular problem arises in decentralized, geographically scattered and non-homogeneous companies. The centralizing of the knowledge engineers would be beneficial for their training and supervision, but it could cause difficulties. The knowledge engineers might be too remote from the experts and local management. They could also be inadequately familiar with the different businesses and, individually, face large amounts of travelling. Superficially everything is much easier in homogeneous and centralized companies except that it is then even more likely that there will be an early battle as to whose "turf" knowledge engineering belongs.

In the event, there have been different organizational practices. Among giant companies the first developments often began in the research departments. Some of these, but by no means all, then emerged as the, de facto, central knowledge engineering section. Depending on their charter and the circumstances of the company they might encourage or allow local activities. Subsequently there arose a need to co-ordinate these activities to arrange for training and to promote corporate policies. These might become the responsibility of a steering committee. If the research department did not itself become involved with the projects of other departments, then somebody else would fill the gap. In the oil companies alone this somebody has included the operational research and the engineering departments.

An alternative pattern has been for DP departments to claim knowledge engineering as their "turf". Occasionally this has been done with great subtlety to combine the strengths of a DP perspective with an openness to the new technology. The strengths include a familiarity with project management, with the needs of programmers, with software interfaces and with program validation. The openness has included a freedom of experts to experiment, the deliberate culling of ideas from outside the company, the launching of joint venture projects to gain experience and the purchase of non DP software (e.g. PROLOG).

On the whole, companies are organizing their projects and their knowledge engineering effectively, in the way that suits the principal participants, but in our opinion:

(a) there is very little evidence of expert systems in vigorous pursuit of the primary corporate objectives;
(b) progress is slow even in companies where there is a competent central team routinely building practical applications;
(c) the audit of knowledge engineering practices and of applications prior to general distribution is still unusual;
(d) insufficient attention and investment has been put into training.

### *4.c Training*

Companies preparing for future developments have been impressed by the academic courses that their staff have attended as background training for their central knowledge engineers. Software suppliers are also considered to give good courses on their products but, apparently, to be less effective on broader topics. In fact there is small regard for the immediate utility of existing, publicly available, courses on knowledge engineering. One cause of this situation is the perception of expert systems in business as being entirely different from those in theoretical studies.

#### *4.c.1 Topics in theoretical training*

Aside from the excessive use of anthropomorphic terms, on courses, the above perception is exemplified by the elicitation of expertise. For years researchers have been solemnly announcing that elicitation is a very, very, difficult task. However, this has not been the case for simpler applications in business, since these applications would otherwise not have been built. In practice, if elicitation is difficult, say, because the expertise is intangible, then that project is avoided. There are always many simpler tasks that could be implemented.

There is a similar disagreement on the details. Several academic courses include a list of warnings about the experts including their forgetfulness, the risk of their being inarticulate and their quickly growing tired. The actual experience is that experts seldom tire of talking about their own skills, are generally very articulate in so doing and will conscientiously use any prompts to their memory that are provided. Preliminary and extensive briefings may be necessary. For the knowledge engineers these will be an introduction into the domain and its jargon so that they do not feel fools. For the experts these will be an introduction into practical expert systems, the reasons for partitioning the task and the necessity to reflect the structure of the expertise. The critical adage seems to be "don't bore the expert or overload the knowledge engineer".

The other main topics typically found in present courses are:

(a) Representation: not an immediate concern in business because most shells rely on rules.

(b) Inferencing and inferencing under uncertainty: uncertainty is only a concern for a minority, but they might gain from courses that dealt with installed shells and their specific inferencing.
(c) Validation: this is always an awkward hurdle but, in practical applications, it is likely to be somewhat similar to the assessment of a human expert.

Finally, courses and books based on systems described in the literature merely fuel the disdain of practitioners. EMYCIN was, for instance, a wholly admirable development in its day but teaching its detail and using it as a current example, in a critical review of shells, is not helpful for studies intended for industry.

#### *4.c.2 Difficulties in practice*

The difficulties regularly mentioned during the study are set out below. It is not suggested that they are more than an input to any new initiative on training.

##### *Gaining and retaining the support of the local manager*

This is the "father" figure to the team mentioned in the 1984 report. Projects continue to be liable to wither away soon after his support is curtailed for any reason (e.g. transfer to new responsibilities). Indeed the transfer of "father" was identified as the chief reason why some large developments failed.

##### *Project management*

This includes the arranging for, the budgetting of and the reporting from the project in a style that fits with other projects in the relevant department. The importance of keeping the paperwork "clean" in large organisations is not to be minimized.

##### *Determining the nature and scope of the application, the necessary expertise and the intended users*

This is obviously critical but may be less portentous than it sounds (e.g. for a tutorial on Fred's experience). Applications are often initially described with becoming modesty. The expertise may have to include modelling and that can be demanding to determine and to achieve. The intended users will often be provisional pending a final review.

##### *Gaining and cultivating the early involvement of a users' representative*

It was encouraging that this item was frequently stressed but, alas, not always achieved because a suitable representative could not be immediately identified, was busy or did not believe in expert systems.

##### *Checking the application is fit for its purpose including any necessary "finishing" for the user*

It is not clear how this checking is normally carried out but, presumably, it is frequently a matter of opinion and there may be a number of people whose opinion is sought by management. The "finishing" required depends on who are the users (e.g. their familiarity with the domain).

In general UK business has relied on apprentice type training, on actual applications, to prepare their knowledge engineers. The Government has recognized apprentice training in some schemes but attendees cannot usually take their expert with them. On-the-job training is very sound for simpler applications but it could prove insufficient for the rapid adoption of future developments.

### *4.d Tools*

The majority of participants in the study expressed no surprise at Johnson's assertion, quoted in the principal findings above, that business applications were dominated by personal computer shells.

No-one appeared keen to challenge it and those working on operational applications appeared rather generous in their assessment of these software products. No quantitative survey has been done but it may be that personal computer shells account for 70–80% of the market in terms of installed products. The rest of the market, which is probably more than 20% by value, consists chiefly of mainframe and minicomputer systems. Work stations and their more elaborate American software are primarily installed for development work within industrial research centres. PROLOG, which is popular in education, appears only to have a minor role in current application but it may find a place in the tool-kit of central knowledge engineers.

In looking ahead there was a sharp divide in the views held. Broadly there were three bodies of opinion:

(a) the heaviest shell users seemed to expect to remain largely with that software and most anticipated:

   personal computers would be the primary machines for both the development and the delivery of applications;
   project building would chiefly be incremental;
   the scope of applications would be gradually broadened;
   better program interfacing and data handling were the major improvements sought in shells.

(b) Those in, or closer to, professional programming groups felt shells were inherently limited and had coloured the choice and depth of applications, they now anticipated:

   alternatives to rules would be essential;
   a greater mix of computers for development and delivery of applications would be employed;
   AI languages associated with programming tool-kits would become more important.

(c) A minority in research departments seemed to believe that there would be a move towards more sophisticated applications, on cheaper work stations, which had overtones of anthropomorphic capabilities.

It is not at all clear which of these views will prove to be most correct. DP departments and some consultants do seem to favour (b) above and their opinion could decide policy in the medium term. However, in the long run the realities of the technology, whatever they may turn out to be, will guide opinion rather than the other way round.

## 5 Personal reflections from the survey

**We believe that:**

(a) the involvement of top management is so vital that it should become a main aim of Government support for this technology and of trade associations plus other institutions seeking a better exploitation of expert systems;
(b) all efforts at promoting the technology, including that of suppliers, should in their own interest seek to reduce the fearful mystery of expert systems and clearly identify realistic benefits achievable in the short term;
(c) project secrecy is a self-imposed wound, even when the application is very rewarding, slowing progress in the very organizations that impose it as well as in others;
(d) training for business needs to be re-thought, preferably jointly with academic teachers in the technology;
(e) applied research programmes should be encouraged in order to seek more practical assistance for current and future applications;
(f) there is little sign of uniformity in the methods used in knowledge engineering, but the practitioners do not regard this as vital as the seeking and achieving of suitable applications;
(g) if it is confirmed that local personal computer shells and AI languages, like extended PROLOG, will grow in national importance, then the future of this software should not be left to small scale suppliers alone.

**Acknowledgement**

The phrasing of this report has been discussed with a number of authorities, including Dr Richard Young and Professor Tom Addis who contributed much to the earlier report. We value highly their input but the responsibility for errors of fact or emphasis is ours alone.

**Appendix A: companies and persons interviewed**

| | |
|---|---|
| BIS Applied Systems | D.G. Hannaford |
| British Coal | R.M.G. Perkin<br>Dr M. Pitt<br>A.E. Price |
| British Gas | C. Ringrose |
| British Petroleum | Dr. W.Q. Limond<br>J.R. Morris<br>P.H. Drake |
| British Telecom | P. Waller<br>L. Morphet<br>N. Dick-Cleland |
| Coopers & Lybrand | C.W. Church<br>Dr. L. Evans |
| Imperial Chemical Industries | M.A.M. Rogers<br>W.H. Swann |
| International Computers Ltd | C.L.F. Haynes |
| Logica | B. Skingle<br>Dr. R. Worden |
| PA Computers & Telecommunications | D.G. Wynd<br>Dr. M.H.M. Turner |
| Peat Marwick McLintock | C. S. Wilmott<br>A. Evans |
| Rolls Royce (Filton) | R. Muir |
| Shell UK | J.F. Pugh<br>I. Ackerley |
| Systems Designers | C. Hebden<br>M. Dulieu |
| Unilever | S. Moralee<br>P. Briggs |

**Appendix B: application reports**

**Report 1**

System: Herald Terminal Fault Diagnosis

Developed by: British Telecom

Herald is one of British Telecom's electronic telephone systems, typically used in offices. The terminals, which exist in great numbers, are complicated electronic telephones containing a printed circuit board. When they go faulty, they are not sent to one of BT's District Repair Centres (DRCs). They are repaired by apprentices (as part of their training) but traditionally under the time-consuming instruction and supervision of experts.

The Solent DRC, situated in Southampton, had heard of the work being carried out by the Advanced Systems Group of BT, which included the building of an expert system shell called Tracker, and saw the possibilities of using expert systems within the DRC. At the same time they were looking for a project on which to test Tracker, and so the two objectives fitted together nicely.

The knowledge for the system was gained from a very experienced expert in the course of seven interviews. These were conducted by two knowledge engineers, one of whom was familiar with the domain, which proved to have useful benefits. Including time spent running through the system as it developed, the total time given by the expert was less than 10 days.

The system was prototyped in sections, these being defined by the design of the circuit. Once all the modules had been prototyped, shown to the expert and changed if necessary, the modules were joined into one knowledge base. At this point the links between the separate modules were put into a system, and a high level faulting and verification procedure was developed.

The knowledge engineer who constructed the system, who had not used the shell or built an expert system before, spent 14 weeks on the project. The knowledge engineer who assisted the interviews spent less than 4 days.

The finished system was tested at two different DRCs, was found to work well, and was also found to be very knowledgeable about the domain.

It is now in use at three DRCs and in one case is enabling an engineer, who is not an expert, to repair the cards, without having to trouble the experts for information or assistance. Being based on sound logical faulting techniques, it has successfully located faults that had not been explicitly entered into the system. Other benefits incude a consistent verification procedure for the repaired circuit boards, reduced costs in terms of man-hours spent and components changed, a wider spread of valuable expertise, and archiving of this expertise. The system also acts as a very good training aid.

The finished system contained 270 rules and over 200 questions and displays, and runs on any IBM PC or compatible with 640K of memory.

**Report 2**

System: Corrosion Prediction (CORPS)

Developed by: British Gas

CORPS is an expert system which assists the corrosion engineer in providing advice at the design stage of a new gas production system. This is a relatively routine but nevertheless time-consuming task which would normally take up about half a day of the expert's valuable time. Using CORPS, however, the same job takes him only a few minutes.

The system is not, however, intended for use by the design team themselves. As the system effectively mimics the way the expert himself would do the job, he is able to appreciate the limitations of the model that is used. He can thus add his own value judgements to the answers before passing them on to the end user.

The corrosion life of the well production tubing depends upon a large number of parameters. These can be divided into, firstly, the physical and chemical characteristics of the well and, secondly, parameters describing the way in which it will be operated. Very often, at the early stage of the design process, only a few of these values will be known. CORPS, however, makes simplifying assumptions wherever information is not available. These are "fail-safe" in nature in that they always provide for the worst possible case. As further, more accurate, information becomes available the system can easily be run again to give progressively more accurate predictions.

In the past, each time the expert was required to advise on the corrosion life implied by a particular set of conditions, it would take him half a day. Sometimes he would have more urgent tasks to do and would not be able to spend the time required to give a fast answer. He can now do so much more easily. He can also encourage the design team to try out different alternatives and compare the results. This means they are able to consider more quickly and effectively the corrosion aspect of the design without demanding large amounts of the expert's time.

CORPS was originally written simply as a demonstration system to try out the expert system shell SAGE. However, the expert which it was attempting to emulate has been impressed with the way it can, by reducing the time it takes, free him for more urgent and intricate tasks without reducing the accuracy of prediction compared to the manual method. Furthermore, the end-users have been impressed by the way their questions can be answered promptly and are able to consider the corrosion life aspect of their task in much more detail. The system is now in regular use.

CORPS has clear benefits in increasing the productivity of the expert, and, by promoting more use of the expert, can only improve the quality of the design of a new production system. This shows that an expert does not have to be large to be useful.

**Report 3**

System: Methane Risk Prediction (UFEL)

Developed by: British Coal

All coal seams contain methane ("firedamp") which is released during mining. The emissions from normal coal-cutting operations can be predicted using algorithmic techniques, and this method has been in use for some years within British Coal.

However, the problem of consistently predicting if mining operations on a new coal face or in driving a new underground roadway will give rise to an emission of methane well beyond the "normal" volume, has never been successfully addressed. The information needed for the prediction is complex, specialized and extensive. Expertise is required from both geologists and ventilation experts, as well as reference to historical data and case histories which indicate where problem areas may arise. Much of the information could be uncertain and some information may not be available for every site.

An application which combines knowledge from several experts is ideal for the expert system approach. The system is called UFEL (Unexpected Firedamp Emission Levels). It was developed in conjunction with Methane Group at Headquarters Technical Department, who provided the domain expertise. UFEL is implemented in the SAGE expert system shell; it is some 6000 lines long, approximating to 700 rules.

The program is based on statistical weightings given to replies from specific questions. From these an overall weighting is derived, which together with production rules is used to give an overall risk assessment. A consultation with UFEL is a thorough process and can take some 20 minutes, which gives an idea of the level of detailed information involved, and indeed why knowledge of this kind is not available except through a knowledge based system.

UFEL starts by making an assessment of the gasiness of the coal mine being considered. Actual measurements can be used if available, otherwise a prediction is made from the geographical location of the mine (within the UK) or by using coal rank and moisture content.

UFEL asks questions on the type, thickness and proximity of nearby coal seams, type and proximity of nearby competent rock beds, fault information and permeability aspects of the strata above and below the workings. Other questions relate to borehole data, nearby old workings, production factors such as coalcutting methods, face stoppages and previous emission data.

UFEL is successfully installed and in use (August 1986) at one coalfield area. It is undergoing validation against specific occurences, and has to date confirmed increased risk factors following three unexpected methane emissions.

Provisional estimates indicate that 2 million tonnes of coal production can be lost per year in a single area through unexpected methane emissions. British Coal operates ten areas, so the potential cost savings are high indeed. The factor of safety cannot be overlooked, and if methane risk prediction using UFEL proves successful then improved safety levels will result. Another important feature of UFEL is its use as a teaching or tutoring aid.

**Report 4**

System: The Advanced Coater Plant Adviser

Developed by: Imperial Chemical Industries plc.

The Advanced Coater Plant (ACP), on ICI's Dumfries site, applies a barrier coating to polypropylene film for use in packaging. The ACP adviser is an expert system which was developed to help operating staff to diagnose and rectify faults on the plant.

As an expert system project, ACP had several advantages. First, fault diagnosis seemed, from the literature, to be well suited to the expert systems approach. Second, the plant itself is reasonably small, and is self-contained rather than being part of a large inter-connected complex, so that the "domain" was well-bounded.

Third, the senior production manager was receptive to new ideas. Finally, the expert, a production technologist, was keen to find a way of spending less of his valuable time trouble-shooting on the plant. He had written a fault guide, but the operators found it hard to follow and did not use it. An expert system might allow him to present his knowledge in a way which staff could understand in his absence.

The first stage of the product was to build a demonstrator system based on a small but non-trivial group of faults. The objective of the demonstrator was to establish whether the problem was soluble and to give the client a guide both to the power of the new technique and to how it would look to its users. The demonstrator was built in one month, using the SAVOIR expert systems shell on an IBM PC XT. The complete system took about 4 months of the knowledge engineer's time (including the demonstrator) and one month of the expert's time, part of which consisted of testing sections of the model and introducing it gradually to its eventual users. The full system contains some 500 "items" and distinguishes between upwards of 80 faults.

The problem was divided into sections corresponding to different groups of symptoms, and initially each section was modelled separately. The analysis was done in separate 1-day sessions, limited to the two per month by the expert's busy schedule. We used the expert's fault-guide as a starting point for discussion. The logic was charted and agreed with the expert, and the charts were programmed directly into SAVOIR. SAVOIR coped well with the requirements for representing the diagnostic knowledge and for controlling the dialogue flow, and programming was rapid. SAVOIR provides flexible windowing and help and explanation facilities which made it possible to engineer a system with an attractive user interface.

As the project went on, the expert progressively refined his perception of the diagnosis problem as a whole. As a result, the original set of conceptually separate sub-systems became increasingly interrelated. This was a challenge to the flexibility of both the software and the knowledge engineer!

The reaction of management and operating staff to the system has been enthusiastic. Several additional benefits have emerged. One is the potential of the system as a training aid. Another is the opportunity to build the standard plant operating data into the system, turning the PC into an "oracle". A third is the ability to make a permanent record of even the rarest fault, which even the expert might forget in time.

**Report 5**

System: Loan Approval Adviser

Developed by: Systems Designers plc.

The Loan Approval Adviser was developed for a Belgian bank (Kredietbank Belgium). It was developed as an addition to the bank's existing system which is based on a simple credit scoring scheme. The objective was to grant loans in the branches more effectively when the credit scoring scheme is inconclusive, without having to involve headquarters.

The Adviser was developed using the SAGE expert system shell running on an IBM-PC-XT. This development was completed in 6 months with a team of two.

The existing Numeric Credit Judgement (NKB) system asks the user all information about the client, the loan and the guarantee. As a result the system produces a scoring list. If the client scores enough points on that list the loan will be awarded. If not, the Loan Approval Adviser system will be started. The expert system also handles those cases which could not be treated by the NKB, such as:

cases where the debt is too high
cases where another signature (as in the case of a married couple the man's or the wife's signature) is needed
cases where the client already has a debt at the bank
a loan for paying back other loans
payments with unequal redemptions or with a currency longer than 60 months

The system asks for more information. After every answer the system draws inferences and decides the next action. The main "goal" is to determine how much confidence can be placed in the client for loan purposes. Based on this confidence rating the system will give advice to the user on whether or not to grant the loan.

The confidence is built up in four main areas.

(a) Confidence in the purpose of the loan, based on currency, amount, advance, target etc.
(b) Confidence in repayment based on income of client, income of partner, income certainty, relation between current cost of living and income, etc.
(c) Confidence in the guarantees, based on what kind of a guarantee the client can offer.
(d) Confidence in the client, based upon the profession, the years of service, the age, the place where he or she lives, if he or she is married, if there are other loans, if the client has other accounts at the bank or at other banks, if he or she is competent, his or her property, experiences in the past, experiences with relatives and the opinion of the clerk of the office where the loan is requested.

Half of all requests can be handled by the NKB system directly (about 500 requests per week). The other half are handled by the expert system. Indications are that more than half of those requests are granted without interference from headquarters. The objective of the project, to relieve the relevant headquarters department, has been achieved.

**Report 6**

Subject: Fire Equipment Planning

Developed by: Shell International

Fireplan is an expert system which determines the fire fighting equipment required for a petroleum tank storage depot.

A depot may comprise up to 15 different types of facility (e.g. fuel tank, road gantry, laboratory) which require protection from fire. The system leads the user through a series of questions which ascertain the size, number and details of each facility. The Safety Code Regulations are interpreted accordingly and results are calculated, for example: the number of foam concentrate tanks; the minimum water pressure; or the number, type and location of fire extinguishers. Printed reports can be produced at several levels of detail.

The system was developed in 3 man months using the micro-based shell "Crystal". The problem was well suited to the clear decision path that Crystal provides and the necessary programming skills were picked up within a week.

Fireplan is now at the Beta Test stage (May 1987). It has been well received by the client so far.

The development process was useful because scrutiny of the Safety Code Regulations revealed some flaws. The system is user friendly and it is perceived that fire fighting equipment will be checked and updated more frequently. This demonstration of expert system capabilities has stimulated ideas for other applications within the client's business area.

**Report 7**

System: Gas Turbine Test Selection

Developed by: Rolls Royce

When changes are made to components within a gas turbine, numbers of tests are carried out to verify and prove that the changes have achieved their planned objectives. This project aims to animate and deliver, by means of an expert system, the manual which advises and guides personnel in the selection of the appropriate tests.

The work is being carried out by the expert largely on his own with only occasional support from a knowledge engineer. The expert had no prior detailed knowledge of computers or expert systems. Xi Plus was chosen as the development vehicle by the expert, with the agreement of the knowledge engineer.

The knowledge is modular (i.e. knowledge for one section of the gas turbine is independent of the knowledge for other sections) and, for any module, exists in layers of increasing depth. Initially, it is proposed to adopt a breadth first development strategy, at a factual level. Later developments will invoke knowledge at heuristic or causal levels, if need be.

The knowledge is partially codified and can readily be represented as facts and production rules.

The system is being developed in phases. The first phase is to develop a prototype to demonstrate the capability and for consultation with the users on the style and sequence of the interaction dialogue. The second phase will develop and deliver successive modules addressing different sections of the gas turbine. It is proposed to embed the system into the company modification procedures at this point. The third phase will develop authoritative documentation to support the modification procedure.

A fully functional prototype for one module has been developed for demonstration to management and users prior to further development.

The Xi Plus shell has been found to be both powerful and friendly. The major facilities which have been used in the prototype include knowledge base chaining, control rules with floating variables, attribute relationships, report files and help files.

**Report 8**

System: DHSS Performance Indicator Analyst

Developed by: Coopers & Lybrand

The DHSS Performance Indicator Analyst constructed by Coopers & Lybrand's Knowledge Engineering Group solved a problem not amenable to conventional computational techniques. It also opened up a new range of expert systems business applications.

Every year the DHSS prepares an annual review of Regional Health Authority performance for presentation to the government minister. Such reviews are increasingly important at times of financial stringency. Some three years ago interdisciplinary committees, consisting of doctors, economists etc. were set up to find a method of analysis for the problems of throughput, efficiency and expenditure that may be experienced by a health authority. They began by considering factors which might lead to problems and isolated 450 performance indicators (PIs), mainly ratios such as doctors per patient, percentage of training nursing staff etc. for which data could be readily collected.

The DHSS used Symphony as a data modelling tool and produced statistical distributions and graphical displays for the PI data. Paper flowcharts were prepared to highlight interesting combinations of PIs. This approach was successful for small discrete areas of the data. However, it soon proved impossible to combine the mass of intermediate conclusions to provide the necessary high-level conclusions for incorporation in the draft ministerial report. Computerization, using the qualitative rules given by the policy experts to produce a textual report, was entirely impossible with Symphony. It was accomplished with an expert system constructed by Coopers & Lybrand using the Crystal shell from Intelligent Environments Ltd.

The Performance Indicator Analyst runs on the IBM PC range and compatibles. It was built in 3 man months to provide an 8000 rule system. Coopers & Lybrand worked with DHSS staff as part of a technology transfer exercise and the system has subsequently been expanded to 11,000 rules by the DHSS themselves.

The PI analyst produces an insightful report on any health authority's performance on the basis of the PIs and relative to the performance of all other health authorities. The expert system is modular both to match the PCs hardware constraints and to reflect the 12 care groups (elderly, psychiatric, etc.) in the DHSS analysis. The largest module has some 1500 rules and 150 kb of text.

The system is unusual in that it has little user interaction. Once the health authority has been selected, the system accesses the data files and produces a screen based and printed report. A complete report on one care group throughout the Regional Health Authority can be produced in about 10 minutes compared with 2–3 hours for a manual analysis. Reports are now produced on an up-to-date ad hoc basis whenever needed in addition to the yearly ministerial requirement.