# Structure, Development and Applications of Expert Systems in Integrated Manufacturing

*B. R. Gaines*

The flexibility, customisation and localisation offered by computer-integrated manufacturing (CIM) are attractive but generate a new class of management problems (Sullivan, 1985; Jaikumar, 1986). The information flows required in manufacturing increase greatly with the diversity of products being handled and the rapidity of change between products (Gershwin et al., 1986). The knowledge required to manage these information flows is not readily available. Setting up a flexible production line requires different and higher level skills than those required to operate it once running. Additional skills are required to design a flexible manufacturing system capable of handling a diversity of products (Gunn, 1981; Morgan, 1984; Kusiak et al., 1985).

Management support systems are needed for the planning, implementation, configuring and operating of integrated manufacturing systems (Elavia, 1985; Miller, 1985; Nassr, 1985). Fortunately, and not as a mere coincidence, new information technologies are becoming available to provide such support. Expert systems provide a technology to cope with the diversity and complexity of the information involved (Kempf, 1985; Liu, 1985; McDonald, 1985; Su, 1986). However, an expert system alone cannot cope with the volume of information about different machine tools and robots, the computation required in grouping, and the provision of numeric simulations for interactive planning and testing. An adequate support system for managing integrated manufacturing systems involves the combination of expert system and simulation-based decision aids with numeric computation, databases, communications and, if it is to be used in operations (Maimon, 1985; Rhodes, 1985; Spur and Mertins, 1985), instrumentation and control systems.

This paper is concerned with expert systems, their essential nature, their development, their role in manufacturing, and their integration with other information technologies and the organisation. The next section gives an overview of expert systems, and is followed by an analysis of the new perspectives and development techniques involved. Later sections cover the roles and interactions of different forms of information technology in supporting the management of a modern production unit.

## Expert systems: An overview

The computer simulation of people in the roles of experts on some topic has become an important application of interactive computer systems. It

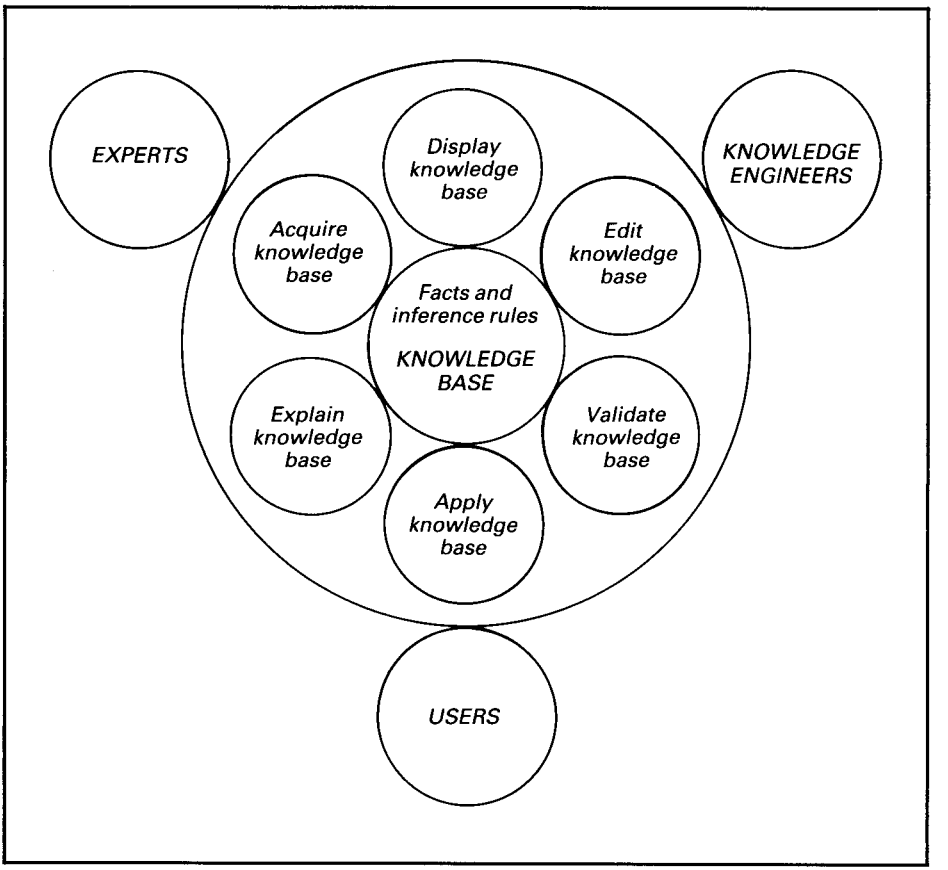


*Fig. 4.1 Structure of an expert system*

has generated a new industry based on creating expert systems to make the practical working knowledge of a human expert in a specific subject area such as medicine or geology widely available to those without direct access to the original expert. Programs now exist that have made practical achievements in medical diagnosis, interpretation of mass spectroscopy results, analysis of geological survey data, and other problems where one would normally go to a human expert for advice (Michie, 1979; Gevarter, 1983; Reitman, 1984). The initial success of expert system developments, together with the development of a number of reasonably domain-independent software support systems for the encoding and application of knowledge (Hayes-Roth et al., 1983; Waterman, 1986), has opened up the possibility of widespread usage. This section gives an overview of expert systems and knowledge engineering.

**Expert system architectures**

Fig. 4.1 shows the basic architecture of an expert system:

- *Knowledge base.* The central ring shows the knowledge base of facts and inference rules that is the heart of an expert system. This extends the facilities of a conventional database by enabling the storage not only of facts but also of rules that enable further facts to be derived.
- *Operators.* The outer ring shows the three classes of operator of expert systems:
  - *users* come to the system for advice on specific problems in a particular domain, supplying it with specific facts and their hypotheses about consequences or goals.
  - *experts* come to the system to transfer their knowledge about the problem domain, supplying it with generally applicable facts and inference procedures.
  - *knowledge engineers* act as intermediaries between the expert and the system, helping him to encode his knowledge and validate the operation of the resultant expert system.
- *Expert system shell.* The middle ring shows the expert system shell, which provides computational facilities for applying the knowledge base to user decision support:
  - the *application system* at the bottom tests the user's hypotheses or seeks to satisfy his goals by deriving the consequences of facts about a particular situation reported by the user, using the general facts and inference rules supplied by the expert and edited by the knowledge engineer.
  - the *explanation system* to the lower left answers queries about the way in which facts have been derived in terms of what information and inference rules have been used.
  - the *acquisition system* to the upper left provides tools for interviewing the expert in order to obtain his vocabulary, and general facts and inference procedures in terms of it.
  - the *display system* at the top provides tools for presenting the knowledge base in an understandable form, and the relations between the facts and inference procedures it contains.
  - the *edit system* to the upper right provides tools for editing the knowledge base while maintaining its integrity in terms of the vocabulary, variables and operations used.
  - the *validation system* to the lower right provides tools for checking the knowledge base against specific case histories with known consequences.

Most current expert system shells:

- Offer a limited knowledge base represented in terms of production rules and frames.
- Have weak application capabilities limited to deriving consequences and testing assertions.
- Give simple explanations in terms of the facts and rules used.
- Have no integrated acquisition capabilities.

- Have limited capabilities to display the knowledge base, sometimes in graphical form.
- Have a user interface to the knowledge base editor which requires understanding of internal data structures.
- Have limited validation capabilities and no integrated database of case histories.

However, rapid improvements are taking place in the knowledge representation, inferencing, explanation, display and editing capabilities of expert systems (Hayes-Roth, 1985; Michalski and Winston, 1985; Pearl, 1986; Swartout, 1986; Rappaport, 1986b). Most current knowledge acquisition is done by manual knowledge elicitation which places too much load on the knowledge engineer and inadequately involves the expert. However, automatic tools for rapid prototyping are becoming increasingly important (Shaw and Gaines, 1983a, 1986b; Gaines and Shaw, 1984b, 1985c,d, 1986e; Boose, 1984, 1985, 1986) and are described in a later section.

**Initial steps in expert systems technology**

One of the first expert system developments was the *fuzzy logic* control system developed in 1974 by Mamdani and Assilian. The system accepted human knowledge of control strategies expressed verbally and encoded it directly as computer programs which acted on the environment (Mamdani and Assilian, 1981). This work was undertaken as part of a study of machine learning in process control and the system controlled was a small steam engine. The verbal rules were of the form shown in Fig. 4.2(a).

Priming the learning machine by such rules was first introduced because learning took too long. It was expected that the rules would act like verbal instructions to a human operator and aid the machine in setting up an initial approximate control strategy. However, it was found that such rules may be interpreted algorithmically to produce computer control programs that are more effective than those derived from classical control theory. Additionally, the learning machine then proceeded only to 'learn' less effective strategies. Hence interest switched to the process whereby human expression of verbal rules that appear vague can lead to highly effective control strategies, i.e. to the development of expert systems.

The rules on which a particular expert system is based may be seen as ways of representing knowledge. This is not a new concept, as such rules have been used in the past quite independently of computer systems. Fig. 4.2(b) is an excerpt from a training manual for lime kiln operators (Peray and Waddell, 1972). It seems that the encoding of expertise into a set of fairly vague but highly applicable rules is a standard way of making it available to others.

In parallel with the controller development, other expert systems were being developed for completely different domains. MYCIN is a

*(a) IF the pressure error is positive and big and the change in pressure error is not negative medium or big,*
*THEN make the heat change negative and big.*

*(b) (1) When the burning zone temperature is drastically low:*
*(i) reduce kiln speed;*
*(ii) reduce fuel.*
*(2) When the burning zone temperature is slightly low:*
*(i) increase ID fan speed;*
*(ii) increase fuel rate.*

*(c) RULE 50*
*IF(1) the infection is primary-bacteremia, and*
*(2) the site of the culture is one of the sterile sites, and*
*(3) the suspected portal of entry of the organism is the gastro-intestinal tract,*
*THEN there is suggestive evidence (.7) that the identity of the organism is bacteroides.*

*Fig. 4.2 Examples of expert system inference rules: (a) rules from a fuzzy logic controller, (b) rules from a lime kiln operator's manual, and (c) rule from MYCIN medical diagnosis system*

well-known medical diagnosis expert system which helps a clinician to act as a consultant on infectious diseases (Shortliffe, 1976). It uses rules of the form shown in Fig. 4.2(c). Such rules are obtained from specialists in microbial infections and their application to particular data is fairly simple data processing. The rules are validated through their application to many cases and revised when they fail to give the correct diagnosis. MYCIN is designed to interact with a clinician in order to make a diagnosis and suggest therapy for a particular patient with suspected microbial infections. It first gathers data about the patient and then uses this to make inferences about the infections and their treatment. Fig. 4.3 shows part of a dialogue between MYCIN and a clinician responsible for testing and treating a patient.

### The spread of expert systems technology

Since the initial success with MYCIN a very wide range of expert systems has been developed. Table 4.1 is a brief summary of some significant expert systems, and Table 4.2 is a brief summary of some major areas of application of expert systems (Gevarter, 1983). The numbers and domains have both increased greatly since these tables were prepared as research into and applications of expert systems have developed worldwide. It is now impossible to make any accurate count since most universities have some activity and many industrial expert systems are regarded as highly proprietary.

### The importance of meta-knowledge

MYCIN works well when the rules are complete and correct, but it was

-------- PATIENT-1--------
(1) Patient's name: (first-last) **FRED SMITH**
(2) Sex: **MALE**
(3) Age: **55**
(4) Have you been able to obtain positive cultures from a site at which Fred Smith has an infection? **YES**
-------- INFECTION-1 --------
(5) What is the infection? **PRIMARY-BACTEREMIA**
(6) Please give the date and approximate time when signs or symptoms of the primary-bacteremia (INFECTION-1) first appeared. (mo/da/yr) **MAY 5, 1975**
The most recent culture associated with the primary-bacteremia (INFECTION-1) will be referred to as:
-------- CULTURE-1 --------
(7) From what site was the specimen for CULTURE-1 taken? **BLOOD**
(8) Please give the date and time when this blood culture (CULTURE-1) was obtained. (mo/da/yr) **MAY 9, 1975**
The first significant organism from this blood culture (CULTURE-1) will be referred to as:
-------- ORGANISM-1 --------
(9) Enter the identity of ORGANISM-1: **UNKNOWN**
(10) Is ORGANISM-1 a rod or coccus (etc.): **ROD**
(11) The gram stain of ORGANISM-1: **GRAMNEG**
(12) Have there been POSITIVE cultures from sites which are NOT associated with infections about which you are seeking advice? **NO**
(13) Did ORGANISM-1 grow aerobically? **YES (8)**
(14) Did ORGANISM-1 grow anaerobically? **YES**
(15) What is the suspected portal of entry of ORGANISM-1 into the sterile site? **GI (6)**

My therapy recommendations will be based on the following probable infection(s) and potential causative organism(s):
INFECTION-1 is PRIMARY-BACTEREMIA
<Item 1> E.COLI [ORGANISM-1]
<Item 2> PSEUDOMONAS-AERUGINOSA [ORGANISM-1]

*Fig. 4.3 Some interaction with MYCIN*

*Table 4.1 Some significant expert systems*

| Expert system | Application |
|---|---|
| DENDRAL | structure of chemical compounds from mass-spectral data and knowledge of mass spectrometry |
| META-DENDRAL | inferring rules of mass spectrometry |
| MYCIN | diagnosis of blood infections and recommendations for treatment |
| TEIRESIAS | knowledge acquisition and debugging |
| EMYCIN | empty (essential) MYCIN expert system shell |
| PUFF | EMYCIN with knowledge base for pulmonary function disorders |
| INTERNIST | advisors for general internal medicine (can deal with symptoms and diagnose 500 diseases) |
| CASNET | diagnosis of glaucoma |
| MOLGEN | scientist's assistant for planning experiments in molecular genetics |
| CRYSALIS | infers protein structure from electron density maps |
| PROSPECTOR | assisting evaluation of geological prospects |
| XCON (or R1) | system for configuring VAXs |
| AM | automated mathematician, discovers interesting conjectures in mathematics |

*Table 4.2 Applications of expert systems*

| *Application area* | *Specific applications* |
|---|---|
| *Analysis* | *electrical circuits, naval threat, mechanics, digital circuits, earthquake damage* |
| *Automatic programming* | *software generation* |
| *Concept formation* | *mathematics* |
| *Control* | *flexible manufacturing, lime kiln, smelting* |
| *Data analysis* | *geology, chemistry, protein crystallography, medicine, oil-well logs* |
| *Design* | *flexible manufacturing, computer system configuration, circuits, chemical synthesis* |
| *Diagnosis* | *medicine, computer faults, nuclear reactor accidents* |
| *Image analysis* | *weather forecasting, medical diagnosis* |
| *Instruction* | *electronics, mathematics, medical diagnosis, coaching, plant operation* |
| *Intelligent assistant* | *weapons assignment, medicine, radiology, computer sales, nuclear plant configuration* |
| *Knowledge acquisition* | *manufacturing, medical diagnosis, geology* |
| *Learning* | *plant diagnosis, chemistry, heuristics* |
| *Management* | *project management, automated factory* |
| *Monitoring* | *industrial plant, patient respiration* |
| *Planning* | *chemical synthesis, robotics, molecular genetics, medical diagnosis, scheduling* |

difficult to set up the rules initially. It was also difficult to trace the rules causing errors in the deductions of MYCIN if the clinician felt that something had gone wrong. To overcome these problems, TEIRESIAS (Davis and Lenat, 1982) was added as an auxiliary expert system with expertise *about* MYCIN. It has the facility to explain MYCIN's decisions and to help the clinician amend the rules when they lead to incorrect conclusions.

TEIRESIAS uses a similar rule-based approach to reasoning as does MYCIN, but the rules are now rules about the forms of rules and the use of rules. The rules of Fig. 4.2 are expressions of inferences from data about the world to conclusions about it, a basic level of knowledge processing. Davis discovered that similar rules could express inferences at a higher level about relations between these basic patterns of inference. A typical example of these *meta-rules* has the form shown in Fig. 4.4(a).

The effect of the meta-rules on the dialogue is that the clinician and MYCIN-TEIRESIAS can discuss the rules and inferences made as well as the patient, organisms and infection. Fig. 4.5 shows a debugging sequence in which the inference made by MYCIN in Fig. 4.3 is noted to be incorrect and TEIRESIAS aids the clinician in correcting it. In this dialogue the clinician noticed an identity was missing in those deduced by MYCIN. He chose to allow TEIRESIAS to guide the debugging, and it used the meta-rule of starting with the identity which appeared incorrectly. The program then reconstructed the deductive process, asking the clinician to confirm its actions at each stage, or comment on

*METARULE 003:*

*(a)* *IF* *(1) there are rules which do not mention the current goal in their premise, and*
*(2) there are rules which mention the current goal in their premise,*
*THEN* *it is definite that the former should be done before the latter.*

*(b)* *IF* *time is small and error is negative big,*
*THEN* *desired change is big.*
*IF* *time is big and error is positive zero,*
*THEN* *desired change is zero.*

*(c)*
*19.* *IF* *concept C possesses some very interesting property lacked by one of its specialisations S,*
*THEN* *both C and S become slightly more interesting.*
*38.* *IF* *there are no known examples for the interesting concept X,*
*THEN* *consider spending some time looking for such examples.*

*Fig. 4.4 Examples of expert system meta-rules: (a) a TEIRESIAS meta-rule, (b) rules from a fuzzy learning controller, and (c) meta-rules of high generality from AM*

anything incorrect. It confirmed the facts about the particular case in hand and checked that the rules were correct. Before very long the issue became sharply focused for the clinician and the system.

The main guideline in debugging MYCIN is that a rule must be found which is relevant to this particular patient, but true in general, and which enhances the set of rules of knowledge. The new rule is added to the set of knowledge rules, with a tag explaining who added it and in what circumstances. This allows the ultimate level of explanation to be given in terms of who added the rule and the reasons given. Whereas MYCIN's rules are specific to microbial infections, those of TEIRESIAS are more general and can be used in other domains. Davis (1983), for example, shows TEIRESIAS being used as an investment decision system for clients of a stockbroker.

### Inductive acquisition of knowledge

The interaction with TEIRESIAS demonstrates the dynamic nature of expert systems. They have often been presented in the literature as a means of capturing expert knowledge as if that knowledge was a static, finished product, universal in nature and owned by one person. In practice knowledge is fluid, situational and generally distributed across groups of people. The initial priming of an expert system with the knowledge of one or more experts is a way of getting started. Meta-rules cannot be effective unless the basic vocabulary and patterns of inference have been at least approximately established. However, an expert system then becomes an entity that grows with the knowledge and experience of its users. It is a support tool for their knowledge acquisition and refinement as well as a way of making it more accessible to themselves and others.

*INFECTION-1 is PRIMARY-BACTEREMIA*
*<Item 1> E.COLI [ORGANISM-1]*
*<Item 2> PSEUDOMONAS-AERUGINOSA [ORGANISM-1]*
*Are all these identities correct?* ***N***
*Are there any identities missing?* ***Y***
*Which one(s)?* ***KLEBSIELLA-PNEUMONIAE PROTEUS-NON-MIRABILIS***
*Are there any identities that shouldn't have appeared?* ***Y***
*Which one(s)?* ***PSEUDOMONAS-AERUGINOSA***
*Do you want to*
*1] track down the problem yourself, or*
*2] allow me to guide the debugging, or*
*3] ignore it for now and fix it another time?*
*Type 1, 2 or 3* ***2***
*We'll start with the PSEUDOMONAS-AERUGINOSA first. The following rules were used in deducing that the identity of ORGANISM-1 is pseudomonas-aeruginosa*
*RULE 184*
*SInce [1.1] the category of ORGANISM-1 is not known*
*[1.2] the gram strain of ORGANISM-1 is gramneg*
*[1.3] the morphology of ORGANISM-1 is rod*
*[1.4] the aerobicity of ORGANISM-1 is facultative*
*There is weakly suggestive evidence (.3) that the identity of ORGANISM-1 is pseudomonas-aeruginosa*
*Is this rule correct?* ***Y***
*Should its premise have failed for this case?* ***Y***
*Which clause of the premise should have been false? [give #]* ***1***
*No conclusions were reached concerning the category of ORGANISM-1. Should it have been possible to conclude a category?* ***Y***
*Which one?* ***ENTEROBACTERIACEAE***
*There were no rules successfully invoked in the attempt to deduce that the category of ORGANISM-1 is enterobacteriaceae.*
*The following (potentially applicable) rules were NOT executed because the starred [**] clause failed:*
*RULE 116*
*If 1) identity of ORGANISM-1 is not known*
*** 2) the gram stain of ORGANISM-1 is not known*
*3) the morphology of ORGANISM-1 is not known*
*4) the site of CULTURE-1 is csf*
*5) the infection is meningitis*
*6) the age (in years) of Albert Hall is less than or equal to .17*
*then there is weakly suggestive evidence (.33) that the category of ORGANISM-1 is enterobacteriaceae*
*RULE 050*
*If 1) the morphology of ORGANISM-1 is rod*
*2) the gram stain of ORGANISM-1 is gramneg*
*3) the aerobicity of ORGANISM-1 is facultative*
*** 4) the infection with ORGANISM-1 was acquired while the patient was hospitalized ***
*then there is suggestive evidence (.7) that the category of ORGANISM-1 is enterobacteriaceae*
*Are all these rules correct?* ***Y***
*Should any of them have been successfully invoked?* ***N***
*Is there a rule missing?****Y***
*Ok, then I need a rule that would allow me to deduce that the category of ORGANISM-1 is enterobacteriaceae.*
*Would you like to give me such a rule now?* ***Y***

*Fig. 4.5 Some interaction with TEIRESIAS*

The use of meta-rules allows expert systems to be programmed interactively by their users. From one perspective TEIRESIAS can be seen as an important development in automatic programming. From another it can be seen as a way in which a machine acquires knowledge through interaction with a person. Meta-rules may also be used to implement knowledge acquisition procedures in which machines acquire knowledge directly through their interaction with the environment – a process of *machine learning* (Michalski and Carbonell, 1983).

Mamdani et al (1976) found that learning could be introduced effectively in a steam engine controller through meta-rules that expressed the way in which the basic rules should be changed as a result of performance feedback. The learning level of their controller operates on rules of the form shown in Fig. 4.4(b).

The use of a very high level of meta-rules for knowledge acquisition is illustrated by AM (Davis and Lenat, 1982), which searches for 'interesting' conjectures in mathematics. What is remarkable about AM is the very notion of what it is doing. This is the highest level of scientific creation, looking for the patterns underlying knowledge. The concept of something being interesting seems peculiarly human and certainly too vague to form the basis of a computer program. However, AM is able to generate not only profound conjectures in mathematics but also meaningful ways of arriving at them. It is judged by the quality of its musings as it considers what it is doing.

Fig. 4.4(c) shows two of the few hundred rules used by AM. Most of them are sufficiently general to apply to other situations. In an organisational context, for example, Rule 19 indicates that a committee which has decision-making powers that do not apply to its subcommittees below a certain size has an interesting property (that of requiring a quorum). Its subcommittees also have an interesting property (that of being quorate).

Inductive inference of knowledge through machine learning procedures is a significant alternative to direct access to the verbal knowledge of experts, and comes closer to capturing the knowledge of main interest. We want to know the basis for what the experts *do* rather than what they *say* they do. Michalski and Chilausky (1980) have reported an interesting comparison of two expert systems for plant disease diagnosis, one constructed directly from the expert's stated rules, and the other constructed automatically by an algorithm for logical induction of structure from behaviour. The expert system with induced rules had a better performance in diagnosis and the expert agreed that its rules, although not obvious at first, were better than his own.

## Features of expert systems

The previous section has shown how applications of expert systems to control and decision systems use linguistic rules, approximate reasoning and meta-rules. It is clear that expert systems use some new computa-

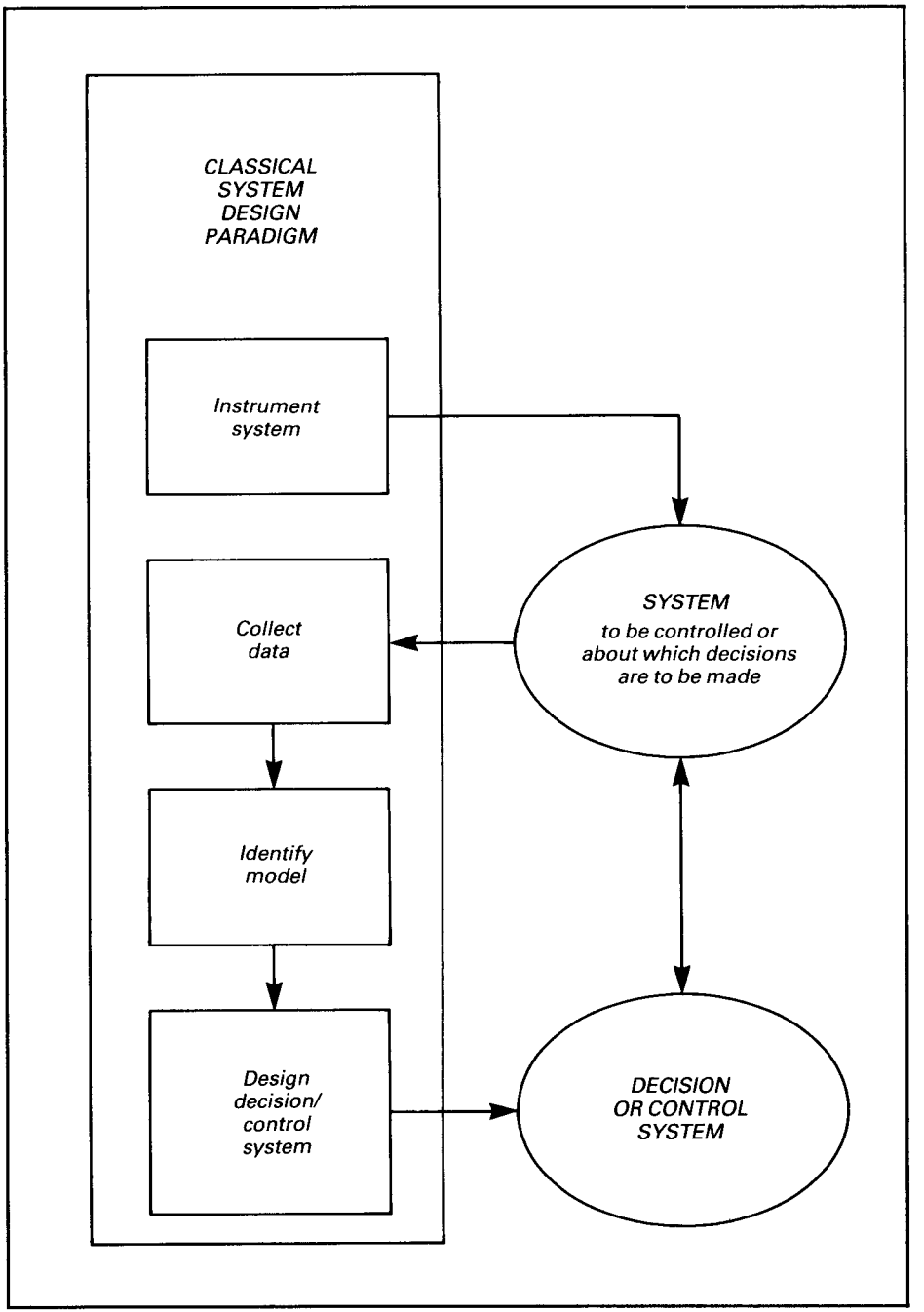


*Fig. 4.6 Classical system design paradigm*

tional techniques, but perhaps less apparent that the system design approach involved is itself fundamentally changed. However, major paradigm shifts are involved (Gaines and Shaw, 1985b). The classical

| | |
|---|---|
| *Classical system design* | *(1) Thoroughly instrument the system to be controlled or about which decisions are to be made*<br>*(2) Use the instrumentation to gather data about the system behaviour under a variety of circumstances*<br>*(3) From this data build a model of the system that accounts for this behaviour*<br>*(4) From this model derive algorithms for decision or control that are optimal in terms of prescribed performance*<br>*(5) Implement the algorithms as a decision or control system* |
| *Problem 1* | *– The models available are inadequate to capture the system* |
| *Old approach* | *– Procrustean Design. Change the world to fit the model – normative technology.* |
| *Paradigm shift* | *– Model Realism. Use system methodologies and information technology that enable the natural world to be modelled without distortion and destruction* |
| *Problem 2* | *– Optimal control is oversensitive to system uncertainties* |
| *Old approach* | *– Suboptimality. Use a suboptimal controller that is robust* |
| *Paradigm shift* | *– Model Uncertainty. Model the uncertainty as part of the system* |
| *Problem 3* | *– Data is unavailable or inadequate for modelling* |
| *Old approach* | *– Managing. Do not automate – leave to human control* |
| *Paradigm shift* | *– Expert Systems. Model the person as a decision maker or controller* |
| *Problem 4* | *– Neither a human nor an automatic system alone is adequate* |
| *Old approach* | *– Ad Hoc System Design. Use a mixture of automatic and human decision/control* |
| *Paradigm shift* | *– Accountable Integration. Integrate automatic and human activity – make the automation accountable – 'why?'* |

*Fig. 4.7 Paradigm shifts in system design*

approach in decision and control system design is the instrumentation, data collection, modelling and optimisation sequence shown in the diagram of Fig. 4.6 and described in the top section of Fig. 4.7. This approach to system design underlies the methodologies of the physical sciences and technologies based on them. It has the merit that it has been extremely successful in engineering much of the technological infrastructure of our current society, including our manufacturing industries.

However, this approach is successful only to the extent that the systems under consideration are amenable to instrumentation and modelling. Its greatest successes have been where this amenability can be achieved normatively, i.e. in cases where the system to be controlled is itself a human artefact. For example, linear system theory has become a major tool in systems engineering, but not because most natural systems are linear – they are not. The implication is in the opposite direction: that linear systems are mathematically tractable and that we design artificial systems to be linear so that we may model them readily.

The application of 'a linear model with quadratic performance criterion' to natural systems is often attempted, but, in general, it does

not work. We have done so not because the tool was appropriate, but because it was the only one we had. However, the use of a hammer to insert screws, although partly effective, tends to distort, destroy and generally defeat the purpose of using a screw. Similarly, the use of an inappropriate system theory to model a system may give useful, but limited, results when we have no other, but it distorts reality, destroys information and generally defeats the purpose of modelling that system.

Much of our current technology succeeds to the extent that it is normative. In agriculture we reduce the complexity of a natural ecology to a comprehensible simplicity by the use of pesticides, herbicides and chemical fertilisers (Gaines and Shaw, 1984c). We reduce the system to one which is amenable to our modelling techniques. That simpler is not necessarily better, and that re-engineering nature to impose uniformity destroys variety which is itself valuable, has only been realised in recent years.

This shift in the paradigm of systems design, from procrustean design to model realism, results in the first perspective shown as a response to Problem 1 in the lower section of Fig. 4.7. Three other changes in perspective stem from this first one.

The second perspective is that which led to new techniques for dealing with uncertainty and approximate reasoning (Gupta et al., 1985). Optimal control theory was regarded as the peak achievement of systems theory in the 1950s and 1960s. However, it proved limited in application because it demanded precision in system modelling that was impossible in practice. It was too sensitive to the nuances of system structure expressed through overprecise system definition.

The third perspective is that which led to the success of linguistic fuzzy controllers and later expert systems. Hayes-Roth (1984) has noted that many of the problems that have been felt to require human management are now amenable to expert systems. Modelling the way the expert performs the task rather than modelling the task itself is the primary characteristic of an expert system.

The fourth perspective is an important one for expert systems. They are knowledge-based systems because they make provision for explaining the decisions reached. The 'why?' question has important implications for both the practical and theoretical significance of expert systems and the logics on which they are based. Expert systems represent new maturity in information technology by providing an audit trail of the data and inferences on which their recommendations are based.

These new perspectives may seem developed for natural systems and less relevant to the artificial systems created for manufacturing. However, the significant parameters of modern manufacturing are determined by its social role in a society of suppliers and consumers. In a recent keynote address, the Executive Vice President of NEC defined manufacturing as the "activity of processing and converting intellectual and physical raw materials and elements into hardware and software

products for ease of consumers' use," and emphasised total manufacturing systems with customers involved as much as employees (Uenohara, 1986). He used the concept of a 'virtual plant' as a worldwide distributed manufacturing process. In the same context the Chief Engineer of GMF Robotics noted that "what is wanted is a new science of manufacturing that integrates technology, management and economics" (Akeel, 1986).

Expert systems, and their theoretical foundations in knowledge science (Gaines, 1986b), are an essential component of a new science of manufacturing.

**Summary**

Expert systems have developed in the past decade from experiments in artificial intelligence to a major commercial industry. They allow human expertise in knowledge representation and inference to be encoded for computers and used for problem-solving. They provide not only a static recording of expertise but also the capability to develop and extend that expertise and tailor it to local situations. Successful applications include not only high-level human cognitive skills such as medical diagnosis, but also important industrial control skills where people carry out tasks that could not be automated using previously available techniques. The most recent developments of expert systems include knowledge acquisition not only through interaction with experts but also through inductive inference from experience. Expert systems technology involves fundamental paradigm shifts in system development and is an essential foundation to a new science of manufacturing.

## Knowledge engineering: Expert systems development

What Feigenbaum (1980) terms *knowledge engineering*, the reduction of a large body of knowledge to a precise set of facts and rules, has already become a major bottleneck impeding the application of expert systems in new domains. We need to understand more about the nature of expertise in itself (Hawkins, 1983) and to be able to apply this knowledge to the elicitation of expertise in specific domains.

The problems of knowledge engineering have been stated clearly:

> "Knowledge acquisition is a bottleneck in the construction of expert systems. The knowledge engineer's job is to act as a go-between to help an expert build a system. Since the knowledge engineer has far less knowledge of the domain than the expert, however, communication problems impede the process of transferring expertise into a program. The vocabulary initially used by the expert to talk about the domain with a novice is often inadequate for problem-solving; thus the knowledge engineer and expert must work together to extend and refine it. One of the most difficult aspects of the knowledge engineer's task is helping the expert to structure the domain knowledge, to identify and formalise the domain concepts". (Hayes-Roth et al., 1983).

This bottleneck is the major impediment to the application of expert systems, and to the realisation of their full industrial potential. The automation of knowledge acquisition and transfer has become a major research and technological goal. This section presents some techniques for the automation of the knowledge engineering task, with examples of their application and indications of their further development and integration.

**Automating knowledge engineering**

The problem of expertise elicitation from a skilled person is well known in the literature of psychology (Nisbett and Wilson, 1977; Broadbent et al., 1986). Hawkins (1983) has analysed the nature of expertise and emphasises its severe limitations and dependence on critical assumptions that are often implicit. Dixon (1981) has surveyed studies showing that much human activity is not accessible to awareness. Collins (1985) has studied knowledge transfer processes among scientists and suggests that some knowledge may not be accessible through the expert, not only because he cannot express it, but also because he may not be aware of its significance to his activity. Bainbridge (1979, 1986) has reviewed the difficulties of verbal debriefing and notes that there is no necessary correlation between verbal reports and mental behaviour, and that many psychologists feel strongly that verbal data are useless. Clinical psychologists see the problem as one of cognitive defences that impede internal communication, and have developed techniques of verbal interaction to identify underlying cognitive processes (Freud, 1914; Kelly, 1955; Rogers, 1967). These can be used to bypass cognitive defences, including those resulting from automisation of skilled behaviour. Hayes-Roth et al. (1983) have given guidelines about knowledge transfer from experts to expert systems, and Welbank (1983) has surveyed the psychological problems of doing this.

In an industrial context the application of techniques derived from clinical psychology to problems of hard system engineering is not an obvious or probable step to take, and the early approach to knowledge engineering was to 'manage' it by inventing the profession of knowledge engineer as defined in the quotation above. It has been possible that knowledge engineering might develop as a profession on a par with systems analysis and programming, and that an initial shortage of skilled knowledge engineers would cause problems to be overcome eventually as the profession developed. However, this scenario now appears less and less likely. There is certainly a shortage of knowledge engineers and problems in developing applications, but doubts have been cast on the notion that human labour is the appropriate solution to the knowledge engineering problem:

- The decline in costs of both hardware and software support for expert systems has brought the technology into a mass-market situation far more rapidly than originally envisioned.

- This has led to a growth in demand for expert systems that is proceeding far more rapidly than the growth in supply of trained and experienced knowledge engineers.
- The declining costs of expert system technology are also making the expense of human labour, in tailoring the technology for particular applications, appear to be the dominating constraint and an excessive cost.
- A move towards a labour-intensive activity such as knowledge engineering is contrary to all trends in industry.
- In particular it is contrary to the trend towards automatic programming techniques in the computing industry.
- The role of the knowledge engineer as an intermediary between the expert and the technology is being questioned not only on cost grounds but also in relation to its effectiveness – knowledge may be lost through the intermediary, and the expert's lack of knowledge of the technology may be less of a detriment than the knowledge engineer's lack of domain knowledge.

These considerations have heightened interest in the possibility of providing knowledge support systems to automate knowledge engineering as a process of direct interaction between domain experts and the computer (Shaw and Gaines, 1983a). Automated knowledge engineering has now been applied in a number of industrial studies of the development of expert systems: to an advisory system for FMS (Gaines, 1986e; Gaines and Shaw, 1986d), to a variety of industrial applications at Boeing Computer Services (Boose, 1984, 1985, 1986), and to studies at Lockheed Software Technology Center (Wahl, 1986).

**Knowledge acquisition for expert systems**

Fig. 4.8 shows the classical system paradigm of Fig. 4.6 extended to include the derivation of design information from the human operator. The obvious approach is interview analysis, and this has been used in many studies. However, often the operator is unable to give a verbal description of his skill, or suggests an incorrect basis for it, and computer modelling of his input-output behaviour gives better results. Textual analysis of the operator's handbook is another significant source of information about his intended behaviour.

These techniques are in fact just part of a hierarchy of knowledge transfer methodologies which defines the areas of application of the knowledge engineering techniques already in use (Gaines, 1986d). Fig. 4.9 shows this hierarchy in relation to the basic architecture of an expert system:

- *Knowledge base.* At the centre, as before, facts and inference rules.
- *Expert system shell.* In the lower oval, the operational system for applying the knowledge base to knowledge support:

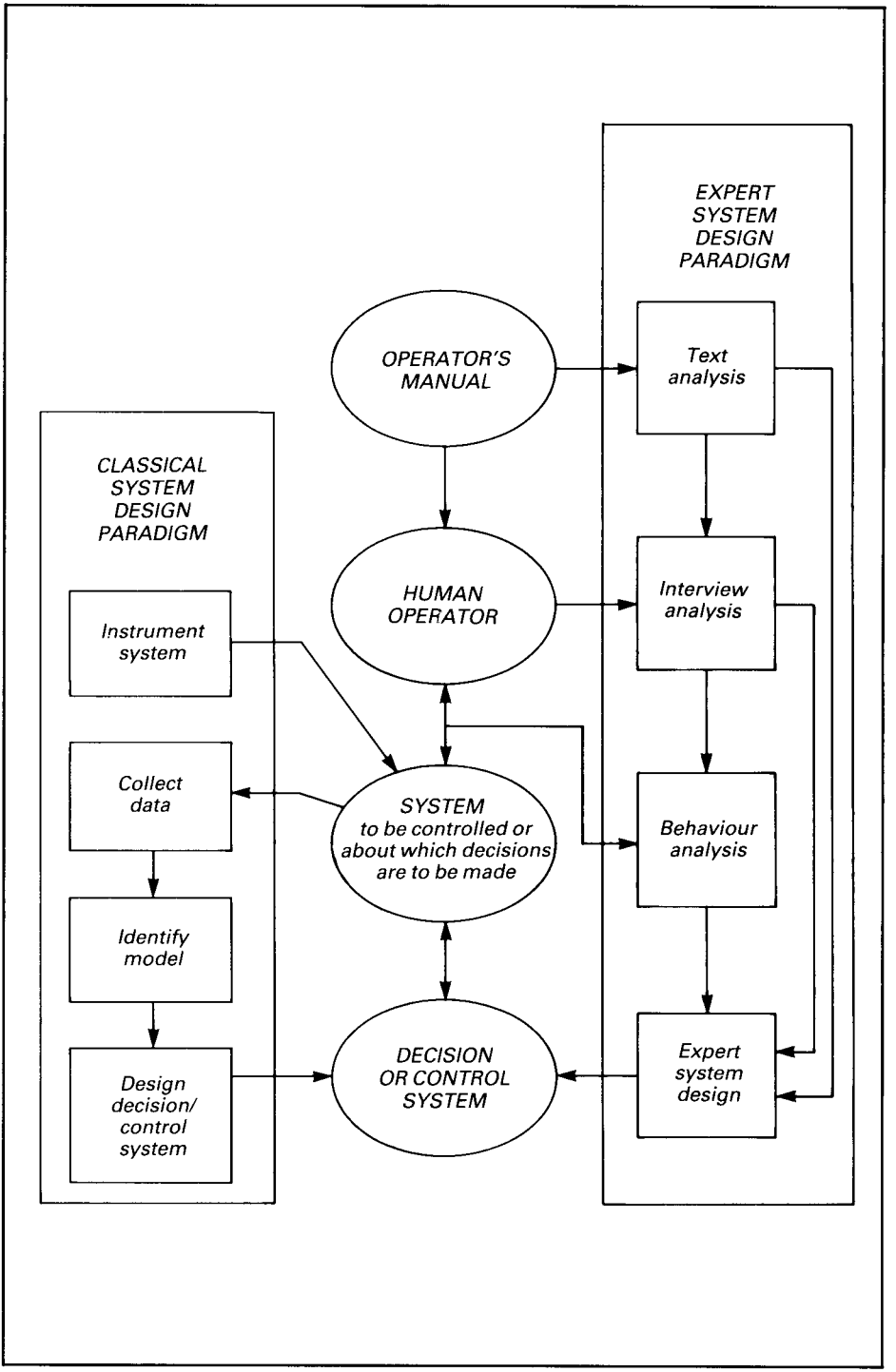


*Fig. 4.8 Expert system design paradigm*

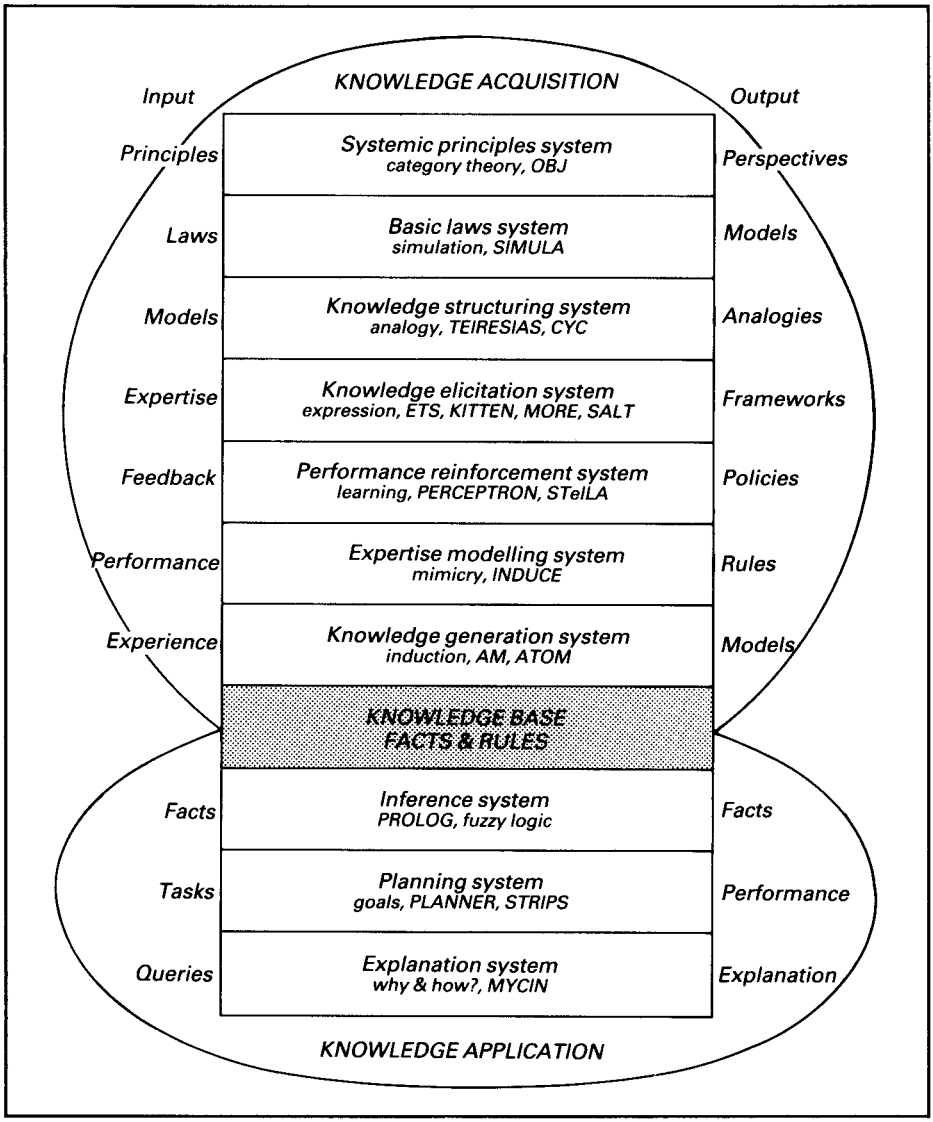


*Fig. 4.9 Hierarchy of knowledge acquisition methodologies*

- the inference system derives the consequences of facts about a particular situation.
- the planning system determines how to use the inference system to satisfy the specified objectives.
- the explanation system answers questions about the basis of the inferences made.

- *Knowledge acquisition system.* In the upper oval, the knowledge transfer processes that may be used to establish the knowledge base:
  - the knowledge generation system implements knowledge acquisition

by the raw induction of models from experience without cultural support, as exemplified by ATOM (Gaines, 1977) and AM (Davis and Lenat, 1982).
- the expertise modelling system implements knowledge acquisition by mimicking an expert's behaviour, as exemplified by INDUCE (Michalski and Chilausky, 1980).
- the performance reinforcement system implements knowledge acquisition from performance feedback, as exemplified by the PERCEPTRON (Rosenblatt, 1958) and STelLA (Gaines and Andreae, 1966).
- the knowledge elicitation system implements knowledge acquisition by interviewing experts, as exemplified by ETS (Boose, 1985), KITTEN (Gaines and Shaw, 1986e), MORE (Kahn et al., 1985) and SALT (Marcus et al., 1985).
- the knowledge structuring system implements knowledge acquisition by analogy, as exemplified by TEIRESIAS (Davis and Lenat, 1982) and CYC (Lenat et al., 1986).
- the basic laws system implements knowledge acquisition by building physical models, as exemplified by simulation languages such as SIMULA (Nygaard and Dahl, 1981).
- the systemic principles system implements knowledge acquisition by derivation from abstract principles, as exemplified by the category-theoretic language OBJ (Goguen and Meseguer, 1983).

The knowledge base and shell have been incorporated with the acquisition methodologies in Fig. 4.9 to show the natural relationships between all three. Shell structure and knowledge acquisition research and development have so far been treated as separate enterprises. Integrating the two is essential to future knowledge-based system applications so that the combined system provides maximal and coherent support to the expert and knowledge engineer.

**Knowledge support systems design considerations**

The general requirements for a knowledge support system to automate knowledge acquisition for expert systems are:

- The knowledge support system tools should be domain-independent.
- The tools should be directly applicable by experts without intermediaries.
- The tools should be able to access a diversity of knowledge sources including text, interviews with experts, and observations of expert behaviour.
- The knowledge support system should be able to encompass a diversity of perspectives including partial or contradictory input from different experts.
- The system should be able to encompass a diversity of forms of knowledge and relationships between knowledge.

- The system should be able to present knowledge from a diversity of sources with clarity as to its derivation, consequences and structural relations.
- Users of the knowledge support system should be able to apply the knowledge in a variety of familiar domains and freely experiment with its implications.
- The system should make provision for validation studies.
- As much of the operation of the system as possible should be founded on well-developed and explicit theories of knowledge acquisition, elicitation and representation.
- As the overall knowledge support develops it should converge to an integrated system.

All of these requirements are subject to caveats: some domain dependence may be appropriate for efficiency in specific knowledge support systems; some human intervention may be helpful or necessary when an expert is using the system; and so on. However, the broad design goals stated capture the key issues in the knowledge support system design community.

The PLANET system for repertory grid elicitation and analysis (Shaw, 1980, 1982; Shaw and Gaines, 1986b) is a primitive knowledge support system satisfying the first two requirements for domain independence and direct use. Its foundations in personal construct psychology, which itself has strong systemic and cognitive science foundations (Gaines and Shaw, 1981a; Shaw and Gaines, 1986a), are attractive in terms of the requirement for theoretical grounding. Boose (1985) in evaluating ETS has noted the limitations of basic repertory grid techniques in terms of the requirement for a diversity of knowledge – that the methodology is better suited for analysis than for synthesis problems (e.g. debugging, diagnosis, interpretation and classification rather than design and planning), and that it is difficult to apply to deep causal knowledge or strategic knowledge – and is attempting to overcome those that use grid hierarchies in NeoETS (Bradshaw and Boose, 1986) and AQUINAS (Boose and Bradshaw, 1986). The TEIRESIAS extension to MYCIN is an early form of knowledge support system providing debugging support for an expert system using basic analogical reasoning (Davis and Lenat, 1982). The development of knowledge support systems has become a major area of activity recently: for example, MORE (Kahn et al., 1985), SALT (Marcus et al., 1985), SEAR (van de Brug et al., 1985) and MOLE (Eshelman and McDermott, 1986).

**KITTEN: A knowledge support system**

Fig. 4.10 shows the structure of KITTEN, an integrated knowledge support system (Gaines and Shaw, 1986e). KITTEN consists of a knowledge base, various analytical tools for building and transforming the knowledge base, and a number of conversational tools for interacting with the knowledge base. The implementation is written in Pascal and

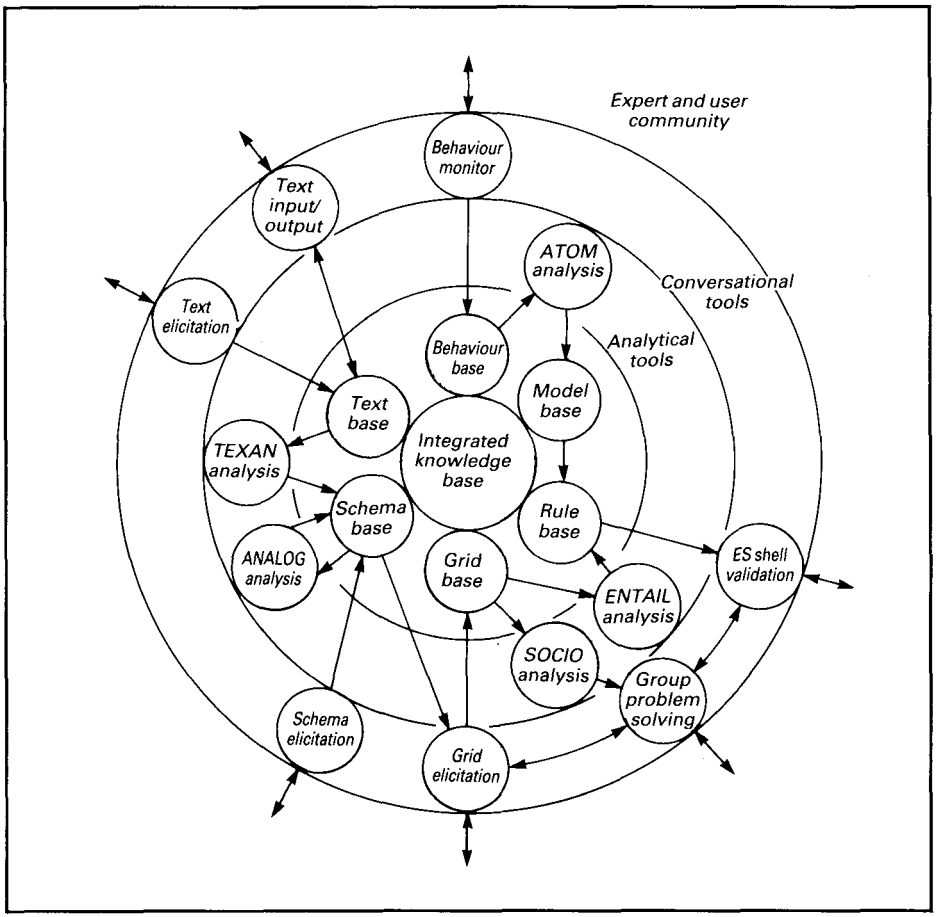


*Fig. 4.10 KITTEN – Knowledge Initiation and Transfer Tools for Experts and Novices*

currently runs on a coupled IBM AT and Apple Macintosh to combine processing power and interactivity.

The KITTEN structure is best understood by following sequences of activity that lead to the generation of a rule base and its loading into an expert system shell. A typical sequence is text input followed by text analysis through TEXAN, which clusters associated words leading to a schema from which the expert can select related elements and initial constructs with which to commence grid elicitation. The resultant grids are analysed by ENTAIL, which induces the underlying knowledge structure as production rules that can be loaded directly into an expert system shell (Gaines and Shaw, 1986c).

An alternative route is to monitor the expert's behaviour through a verbal protocol, giving information used and decisions resulting, and analyse this through ATOM, which induces structure from behaviour and again generates production rules (Gaines, 1977).

These two routes can be combined. KITTEN attempts to make each stage as explicit as possible, and, in particular, to make the rule base accessible as natural textual statements rather than technical production rules. The expert system shell being used in KITTEN currently is Nexpert (Roy, 1986; Rappaport, 1986a,b), which gives a variety of textual and graphical presentations of the rule base, enabling the expert to see the impact of different fragments of knowledge.

The group problem-solving component of KITTEN is particularly important because it goes beyond the stereotype of an 'expert' and 'users', and allows the system to be used to support an interactive community in their acquisition and transfer of knowledge and mutual understanding. The SOCIO analysis allows members of a community to explore their agreement and understanding with other members, and to make overt the knowledge network involved (Shaw, 1980, 1981).

The KITTEN implementation is an initial knowledge support system prototype offering a workbench with minimal integration of the knowledge base. Each of the tools included has already proven effective, and their combination is proving very powerful in stimulating experts to think of the knowledge externalisation process from a number of different perspectives.

The following subsections describe and illustrate one of the major knowledge elicitation tools in the KITTEN knowledge support system.

**Repertory grid elicitation of knowledge**

The elicitation of knowledge from experts through general, domain-independent techniques depends on the availability of a framework within which to represent knowledge. This framework needs to be well founded in terms of both cognitive psychology and computer science. The *repertory grid* is a form of knowledge representation based on a systemic theory of human cognition founded on the single primitive of a *construct*, or dichotomous distinction (Gaines and Shaw, 1981a, 1984b, 1985c; Shaw and Gaines, 1981b,c, 1983b, 1984). For an individual, constructs are "transparent template which he creates and then attempts to fit over the realities of which the world is composed" (Kelly, 1955).

All of human activity can be seen as a process of anticipating the future by construing the replication of events: "Constructs are used for predictions of things to come, and the world keeps rolling on and revealing these predictions to be either correct or misleading. This fact provides a basis for the revision of constructs and, eventually, of whole construct systems" (Kelly, 1955).

The personal construct psychology model of man is strongly epistemological and concerned with the way in which man models his experience and uses this model to anticipate the future. The anticipation may be passive, as in prediction, or active, as in action.

Knowledge engineering may be seen as eliciting the constructs of an expert in the relevant subject area. The most common method used for

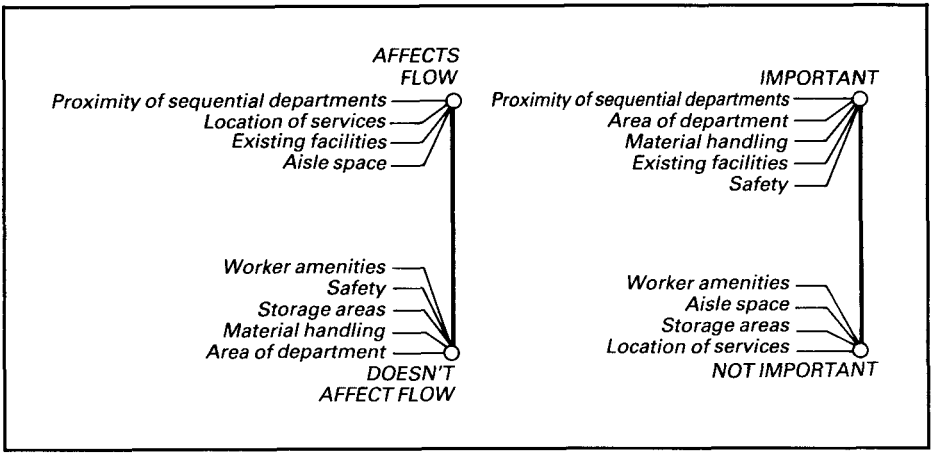


*Fig. 4.11 Binary constructs elicited about factory layout*

eliciting a construct is the minimal context form or triad method. The elements are presented in groups of three, three being the least number which will produce both a similarity and a difference, and the subject is asked to say in what way two are alike and thereby different from the third. This is the emergent pole of the construct. The implicit pole may be elicited by the difference method (in what way does the singleton differ from the pair?) or by the opposite method (what would be the opposite of the description of the pair?).

As an example, take three factors affecting factory layout (Adler and Matheson, 1986), *location of services, aisle space* and *area of department.* In what way are two alike and thereby different from the other one? We might first of all say that *location of services* and *aisle space* are alike since they *affect flow*, whereas *area of department* does not. This is, then, the first construct with its two poles or opposite descriptions. Now all the elements in the set must be rated on this dimension as being either *affects flow* or *doesn't affect flow,* as shown on the left side of Fig. 4.11. This also shows the significance of the term 'personal' in personal construct, since it would not necessarily be a publicly agreed description that safety doesn't affect flow.

Then the second and subsequent constructs are elicited in exactly the same way using different triads each time. The right side of Fig. 4.11 shows a further construct elicited. This second construct illustrates that constructs can be factual, imaginary, emotional or whatever is important to the person generating the grid.

A scale allowing more distinctions than the pair 1 and 2 may be used as required. If a 1 to 5 scale is used then the above example might become the grid shown in Fig. 4.12. Note how the use of a multi-point scale with an odd number of values allows for a central rating, in this case 3, which does not force the user to choose either pole. It may be desirable to apply more discrimination to this central rating and allow the subject the choice

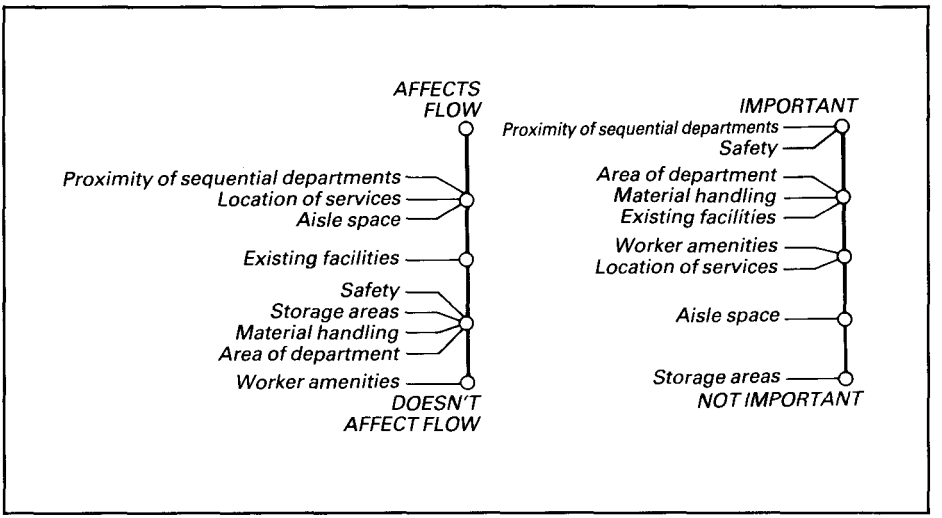


*Fig. 4.12 Rating scale constructs elicited about factory layout*

of the two possibilities: neither, i.e. the element belongs to neither pole; or both, i.e. the element belongs to both poles (Landfield, 1976). It is also possible to extend these possibilities to allow separate rating on each pole (Shaw and Gaines, 1980; Gaines and Shaw, 1981b).

**Automated repertory grid elicitation and analysis**

In the past the elicitation of repertory grids has been a task requiring a skilled psychologist who can draw out a person's constructs whilst not imposing his own ideas or personality. This has involved the development of a variety of techniques and methodologies for grid elicitation (Kelly, 1955; Fransella and Bannister, 1977). The coming of powerful, low-cost personal computers has made it possible to program some of the elicitation techniques used by people in a form where they can be applied automatically. It is now possible to codify some of the techniques used by people in a form where they can be administered through a conversational interactive computer program.

PLANET (Shaw, 1982; Shaw and Gaines, 1986b) and ETS (Boose, 1985) access the expert's personal construct system by interactively eliciting a repertory grid of constructs classifying elements characteristic to part of the domain of expertise. They both use interactive prompt-response sequences to elicit the expert's ratings of elements on constructs. In KITTEN, constructs are elicited by using the natural click-and-drag techniques of the window-mouse interaction rather than numeric rating scales. Fig. 4.13 shows a screen implementing triadic elicitation being used in a study of management effectiveness. Fig. 4.14 shows the elicitation of the pole name for the similarity between the selected elements. Fig. 4.15 shows the remaining elements being rated by dragging them to the scale.

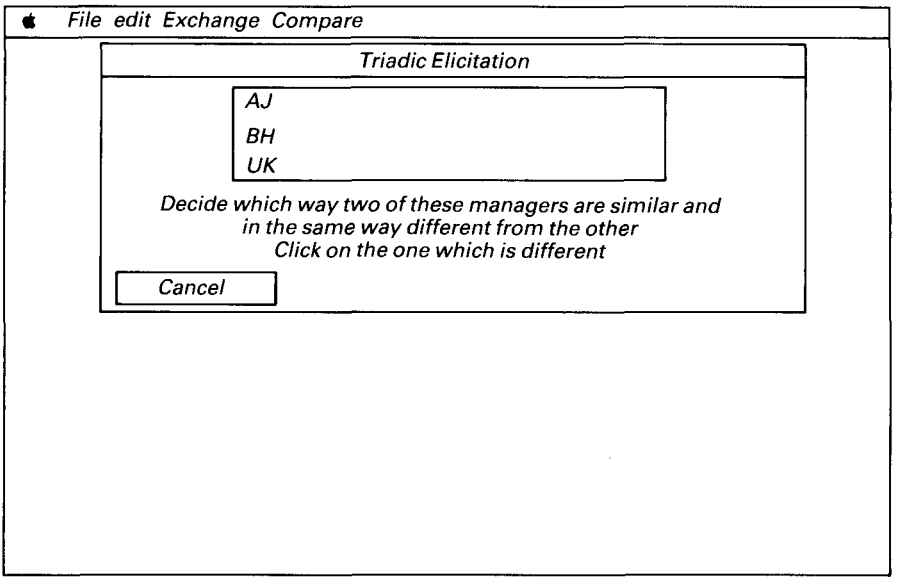


*Fig. 4.13 Interactive triadic elicitation of constructs*

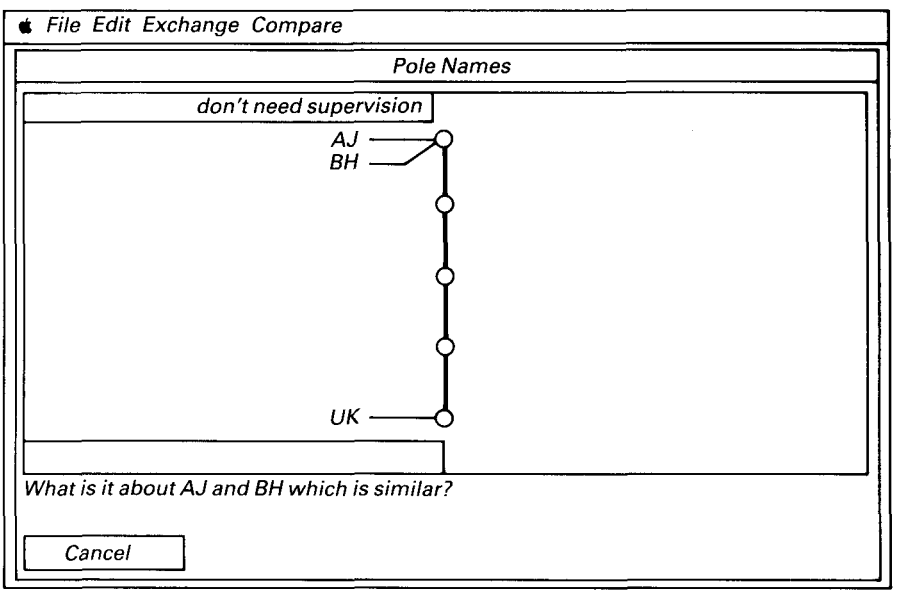


*Fig. 4.14 Interactive entry of pole names for constructs*

An important feature of interactive grid elicitation is that on-line analysis may be used to give feedback to the expert about his construct system, e.g. that two elements have not been adequately discriminated, or that two constructs are giving very similar discriminations. This feedback may be used to elicit additional constructs and elements. Fig. 4.16 shows the natural representation of the construct match screen supporting element elicitation in KITTEN.

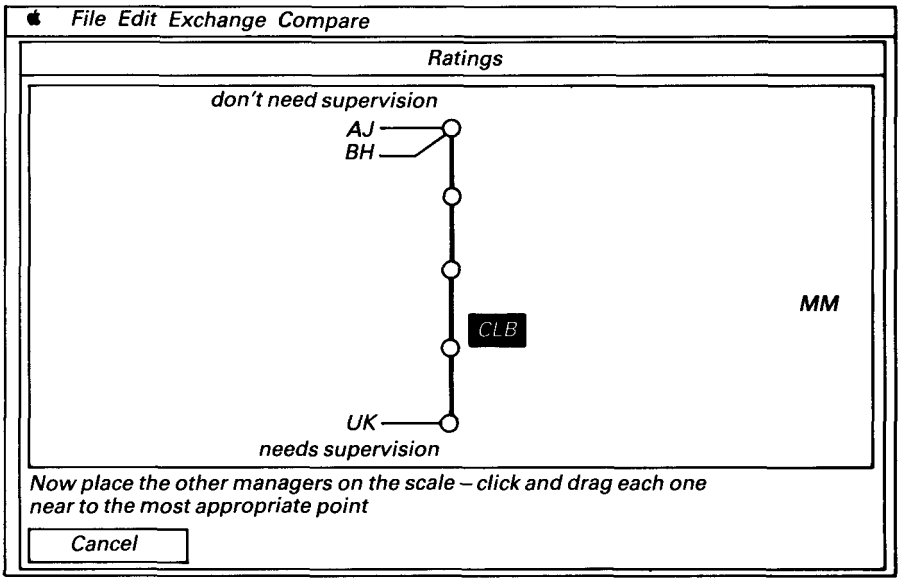


*Fig. 4.15 Click-and-drag rating of elements on constructs*

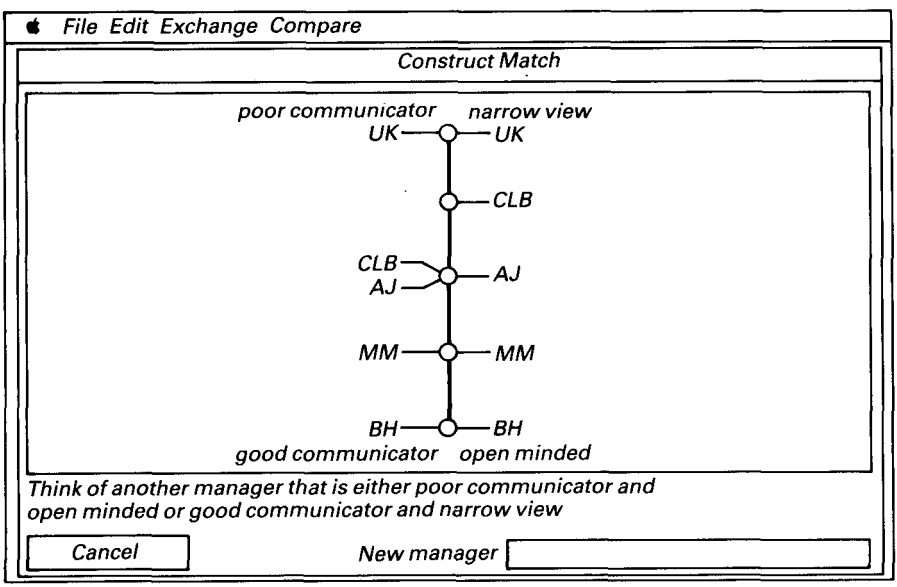


*Fig. 4.16 Element elicitation through a construct match*

It is important that knowledge elicitation methodologies cope with a group of experts as well as the individual. Much expertise resides only within the social context of cooperating individuals and requires elicitation across the group (Boose, 1986; Kernaghan and Cooke, 1986). KITTEN provides a number of facilities for group interaction using the *participant systems* approach proposed by Chang (1986a,b) and implemented in a Macintosh network (Shaw and Chang, 1986; Shaw,

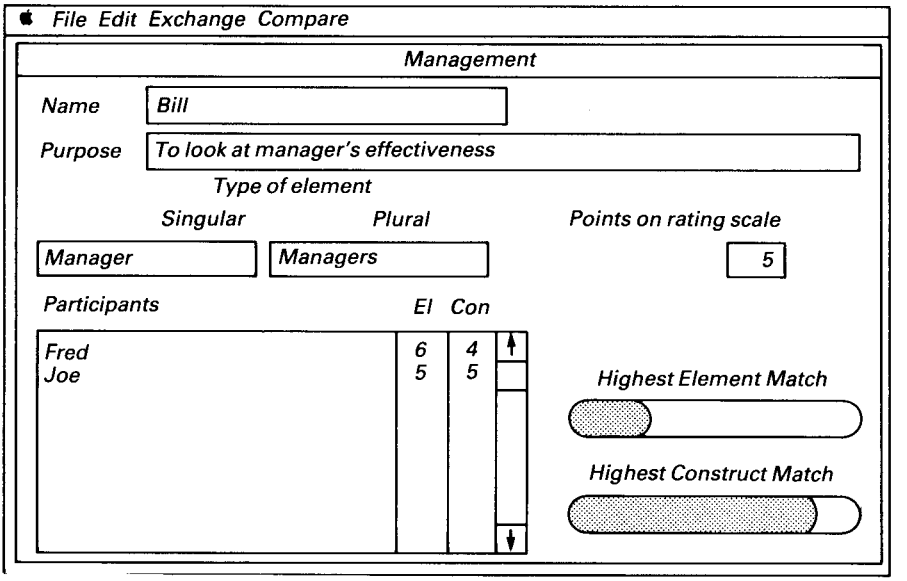


*Fig. 4.17 Status of elicitation system for a group of participants*

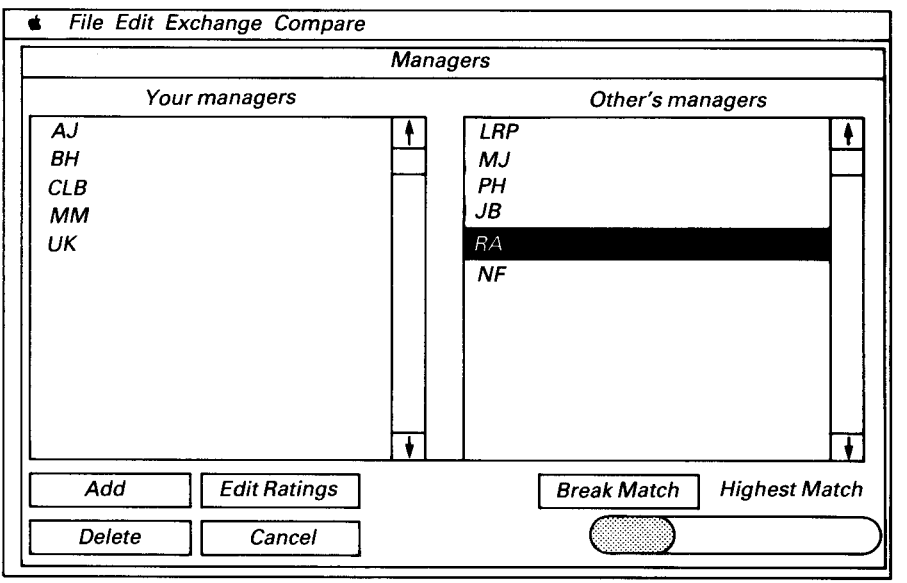


*Fig. 4.18 Selecting another participant's elements to add to one's own*

1986). Fig. 4.17 shows a status screen allowing users to see how many participants are involved in the elicitation, how many elements and constructs they are using, and so on. A user can respond, for example, to the request for a new element shown in Fig. 4.16 by transferring to the screen shown in Fig. 4.18 and choosing to add an element entered by another participant into his own set.

The SOCIO programs in KITTEN support group elicitation techni-

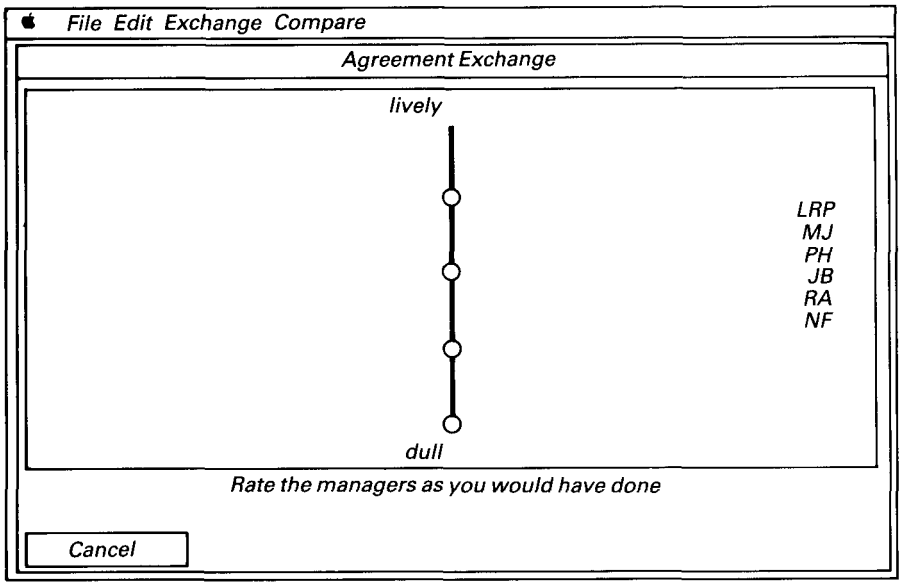


*Fig. 4.19 Rating elements on another participant's constructs*

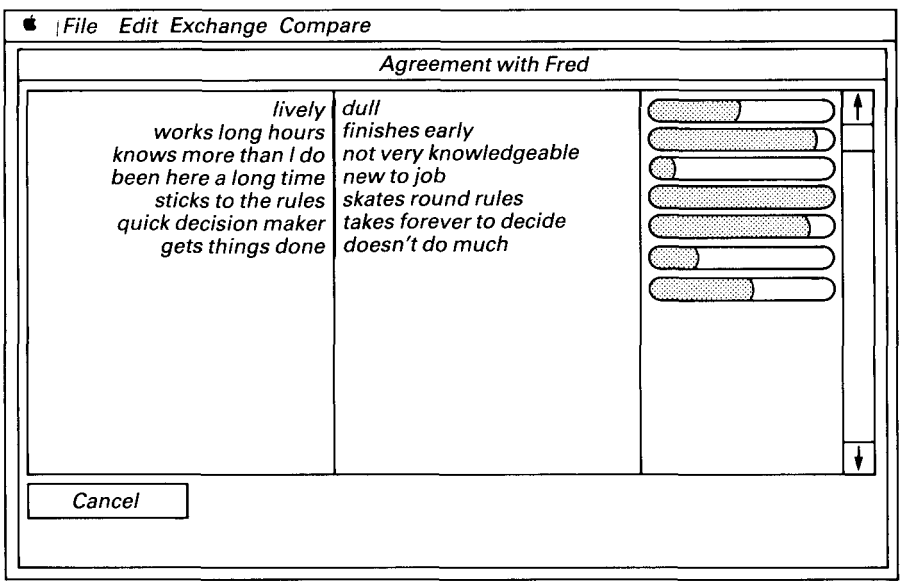


*Fig. 4.20 Presentation of degrees of agreement*

ques in which the construct systems of a number of users are compared. Grids are elicited separately but then exchanged in two ways: a user can place elements on a colleague's constructs from his own point of view, and the analysis system then allows him to explore their agreement; or he can attempt to place them from his colleague's point of view and hence explore his understanding. Fig. 4.19 shows the screen when a participant is being asked to rate the elements using another participant's constructs,

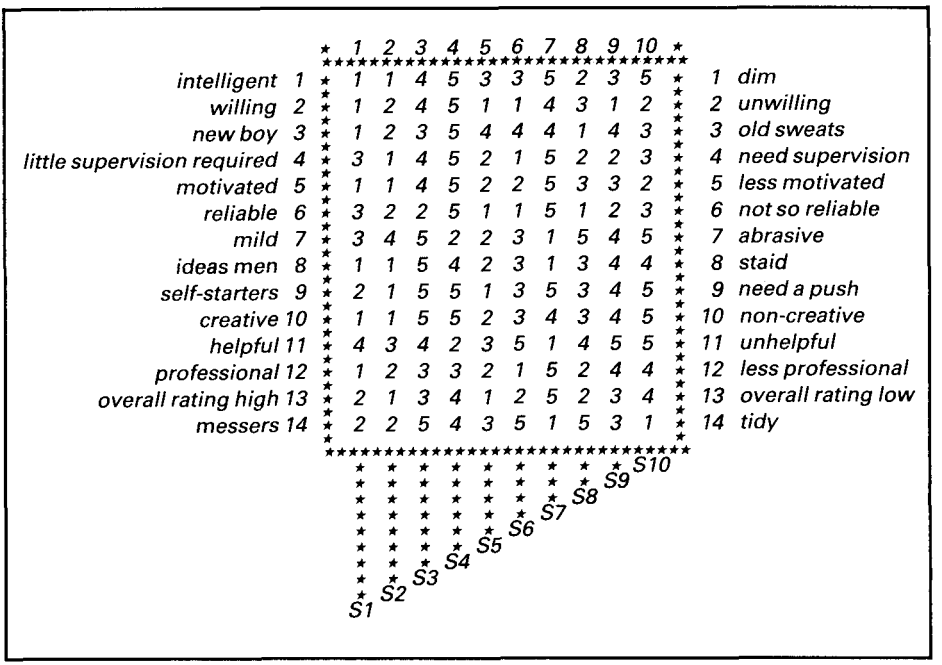


*Fig. 4.21 A repertory grid on staff appraisal*

and Fig. 4.20 shows the presentation of the degree of agreement when this has been done. Such techniques allow the similarities and differences in expert perspectives to be rapidly identified.

**Automated rule generation from repertory grids**

Once knowledge is represented in a repertory grid it may be analysed to generate rules suitable for priming an expert system shell. The *entailment analysis* of a repertory grid treats each pole of a construct as a fuzzy predicate to which the elements have degrees of membership given by their ratings, and induces the logical implications between these predicates (Gaines and Shaw, 1981b). ENTAIL is a program that ranks order entailments in terms of the uncertainty reduction they induce in the distribution of the data, and hence tends to reject spurious rules (Gaines and Shaw, 1986c).

Fig. 4.21 shows a fully developed grid on personnel selection represented in numeric form for use as a data structure for computer processing. A repertory grid may be viewed as a component of a database in entity-attribute form with elements as entities, and constructs as attributes and ratings of elements on poles of constructs as values. Fig. 4.22 is an ENTAIL analysis of the grid of Fig. 4.21 with the truth values and uncertainty reductions (termed 'information') shown together with the entailments expressed linguistically. The uncertainty reduction measures the significance of the hypothesis and is used to ensure that

| Entail | Truth | Inf. | (Cutoff 0.18) Implication |
|---|---|---|---|
| L 1 → L 9 | 0.75 | 0.29 | intelligent→self starters |
| L 9 → L13 | 1.00 | 0.29 | self starters→overall rating high |
| R 9 → R 1 | 0.75 | 0.28 | need a push→dim |
| L10 → L 8 | 1.00 | 0.28 | creative→ideas men |
| L 1 → L10 | 0.75 | 0.26 | intelligent→creative |
| R 8 → R10 | 1.00 | 0.26 | staid→non-creative |
| L10 → L 9 | 0.75 | 0.26 | creative→self starters |
| R13 → R 6 | 0.75 | 0.26 | overall rating low→not so reliable |
| L 9 → L10 | 0.75 | 0.24 | self starters→creative |
| R10 → R 1 | 0.75 | 0.24 | non-creative→dim |
| L10 → L 1 | 0.75 | 0.23 | creative→intelligent |
| R13 → R 9 | 1.00 | 0.23 | overall rating low→need a push |
| R 4 → R13 | 0.75 | 0.22 | need supervision→overall rating low |
| R 5 → R 4 | 0.75 | 0.22 | less motivated→need supervision |
| R 5 → R13 | 0.75 | 0.22 | less motivated→overall rating low |
| R 9 → R10 | 0.75 | 0.22 | need a push→non-creative |
| L 1 → L 3 | 0.75 | 0.21 | intelligent→new boy |
| L 6 → L13 | 0.75 | 0.21 | reliable→overall rating high |
| R10 → R 9 | 0.75 | 0.20 | non-creative→need a push |
| R 1 → R 6 | 0.50 | 0.19 | dim→not so reliable |
| R 1 → R10 | 0.75 | 0.19 | dim→non-creative |
| R 9 → R 4 | 0.50 | 0.19 | need a push→need supervision |
| R 9 → R12 | 0.50 | 0.19 | need a push→less professional |
| R 9 → R13 | 0.50 | 0.19 | need a push→overall rating low |
| R12 → R13 | 0.75 | 0.19 | less professional→overall rating low |
| R13 → R 4 | 0.75 | 0.19 | overall rating low→need supervision |
| R13 → R12 | 0.75 | 0.19 | overall rating low→less professional |
| L 4 → L 5 | 0.75 | 0.18 | little supervision required→motivated |
| L 4 → L 9 | 0.50 | 0.18 | little supervision required→self starters |
| R 6 → R 4 | 0.75 | 0.18 | not so reliable→need supervision |
| R 6 → R13 | 0.75 | 0.18 | not so reliable→overall rating low |
| L12 → L 9 | 0.50 | 0.18 | professional→self starters |
| L13 → L 4 | 0.75 | 0.18 | overall rating high→little supervision required |
| L13 → L 5 | 0.75 | 0.18 | overall rating high→motivated |
| L13 → L 9 | 0.50 | 0.18 | overall rating high→self starters |

*Fig. 4.22 Inference rules derived from the repertory grid on staff appraisal*

trivial entailments consistent with the data are pruned. Note how a hypothesis of high truth value such as R13→R9 can give less information reduction than one of lower truth value such as L1→L9.

The entailments derived by ENTAIL can be converted directly to rules that may be loaded into a standard expert system shell. The truth value of the rule codes into the truth value representation used by the shell. The rules may be checked by the expert for their content, but their prior evaluation for uncertainty reduction makes it unlikely that spurious rules will be generated. The rules form a decision tree expressing the relationship between an expert's data and his conclusions, both personally expressed. For example, the data of Fig. 4.21 may be regarded as that of an expert on staff appraisal and used to derive his *overall rating* (construct 13) from behavioural assessments such as *intelligent* and *creative*. The ENTAIL analysis of Fig. 4.22 shows that L1, L4, L6, L9, L10 and L12 imply L13, i.e. that 'intelligent, creative, reliable and professional self-starters requiring little supervision receive a high overall rating', whereas R1, R4, R5, R6, R8, R9, R10 and R12 imply R13, i.e.

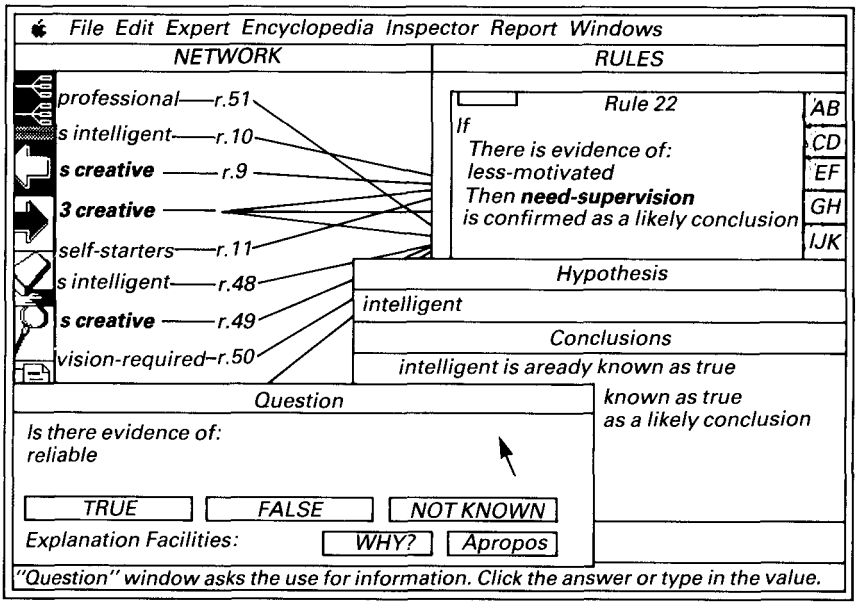


*Fig. 4.23 Inference rules derived from the repertory grid on staff appraisal operating in the Nexpert expert system shell*

that 'being dim, needing supervision, less motivated, not so reliable, staid, needing a push, or less professional leads to a low overall rating.'

Fig. 4.23 shows the expert system shell Nexpert in operation loaded with the entailments of Fig. 4.22 used as inference rules. Interaction with Nexpert enables the expert to see the derived rules in action. He can determine their consequences with test data, analyse new hypothetical cases, and see the interrelations between rules presented graphically. The logging and explanation facilities of Nexpert enable him to track down spurious inferences that may arise with the rules derived by ENTAIL, or proper inferences that are missing. He can then edit the rules and test the revised system using Nexpert's facilities.

KITTEN provides many other techniques for the analysis of individual and multiple repertory grids, for the development of conceptual schema, and for text and behaviour analysis. It allows all the expert system design techniques on the right of Fig. 4.8 to be used in an integrated system.

## Summary

This section has argued that, while knowledge engineering is certainly the area to emphasise in the commercial exploitation of expert systems technology, it is misleading to assume that it necessarily involves the detailed involvement of knowledge engineers. Knowledge engineering can, and should, be automated through knowledge support systems that are used directly by the subject domain experts. The knowledge engineer should operate only at the meta-level in helping the expert to select and

apply the tools within these knowledge support systems. This has a number of significant advantages:

- Efficient transfer of expertise without the overhead of intermediaries.
- Accurate transfer of expertise without distortion by intermediaries.
- Increased awareness by experts of the nature of their knowledge and expertise.

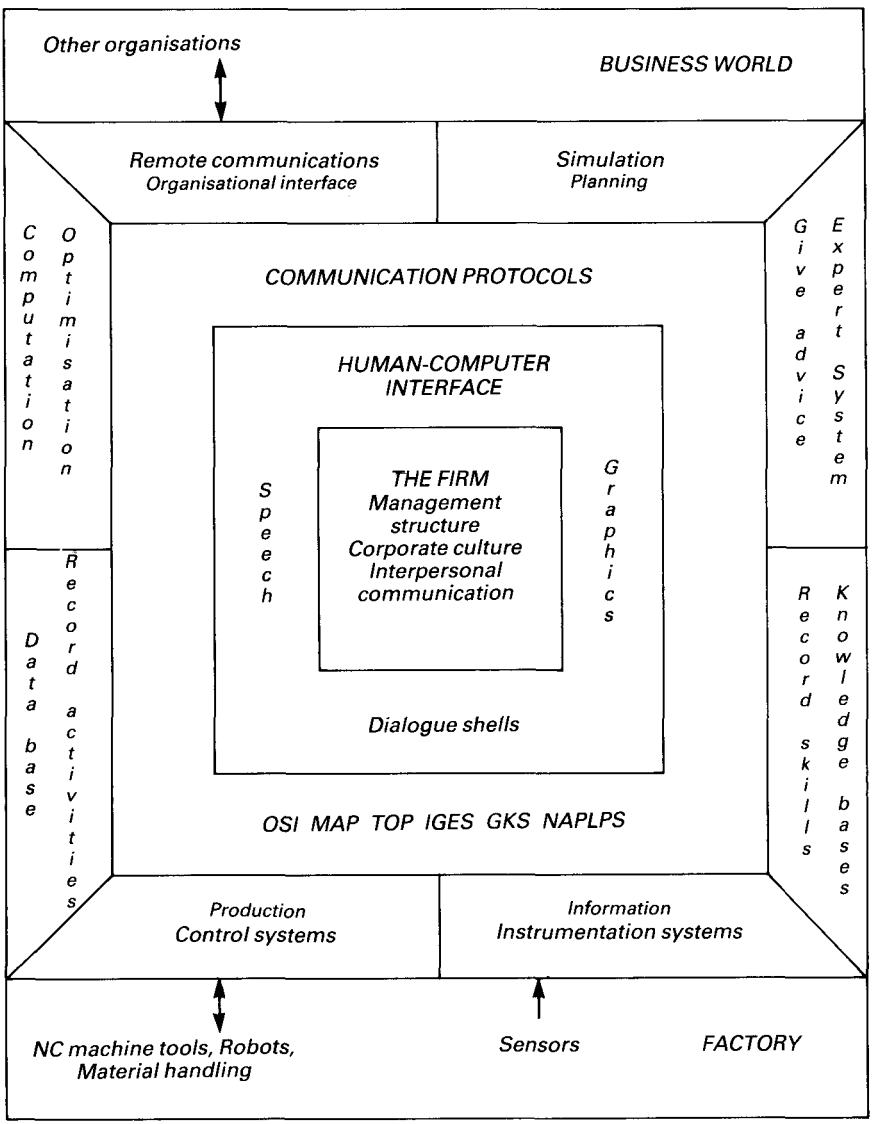


*Fig. 4.24 Information technologies involved in CIM*

A number of practical knowledge engineering tools have been described which illustrate various possibilities in automating knowledge engineering. Some of these tools have already been used widely for rapid prototyping of industrial expert systems; others are experimental, illustrating the frontiers of current research. The integration of such tools into versatile knowledge support systems is a major research thrust of a number of organisations concerned with expert systems development. Any project involved in expert systems applications should monitor the development and availability of such tools, and regard them as a basic technology of at least equal importance to those of expert system shells.

## Information systems in integrated manufacturing

Expert systems provide a significant new technology for integrated manufacturing systems, but they form only part of the spectrum of information technologies essential to CIM. Fig. 4.24 shows the information systems in a modern production unit using CIM techniques. The outer ring comprises eight basic information technologies that together support the integrated manufacturing system and its interface to the factory and business world. The centre shows the firm itself with its corporate structure, and the rings surrounding it show the human interfaces and communication protocols necessary for the systems' management and integrated operation. The figure has been simplified to the minimum structure necessary to support an effective integrated manufacturing system, and the residual complexity shows the need to consider the total context of CIM technology. Such a system will not necessarily succeed if any one component, or even several, are advanced state-of-the-art technologies. However, it will certainly fail if one component is inadequate or the total system is not well integrated.

### The firm

The central core is the non-technology part of the system, the people that manage and operate the plant, their organisation and value systems. The overt operation of the integrated manufacturing system is part of the *technical culture* of the firm in Hall's (1959) terms. It can be communicated, documented and partly programmed. It can also be changed fairly readily in response to new requirements, techniques and technologies. The management structure of the firm is part of its *formal culture*. Only part of it is overt and documented, and it is resistant to change. What has come to be called the 'corporate culture' (Deal and Kennedy, 1982) is in Hall's terms the *informal culture* of the firm. It pervades the organisation, is communicated by example, rarely documented, and changes only slowly or under pressures of survival.

The managerial structures and social processes of the firm are not normally treated as part of its information technology. However, it is possible to develop socio-cognitive models of organisations that encom-

pass both the people and the information technology supporting them (Shaw and Gaines, 1981a, 1986a), and Hall's cultural model is significant to knowledge engineering for expert systems. In particular, in order to be successful the technical systems must be consistent with the organisational structures and processes. This applies particularly to interpersonal communications where computer networks will not be used if they violate the managerial and social norms of the firm, and to decision and control paths which have to conform to existing authority and procedures that are not necessarily those of the organisation chart.

### Human-computer interface

Recognition that the managerial and social structure of the firm dominates its style of operation draws attention to human-computer interaction as a critical technology. Advances in graphics, voice input/output and intelligent terminals have improved the technical repertoire for human-computer interaction (Gaines and Shaw, 1984a; Shaw and Gaines, 1985). Widespread contact with low-cost personal computers has increased computer literacy and user acceptance of the technology. The human factors are being studied of CAD (Majchrzak et al., 1987) and FMS (Mitchell et al., 1984). However, much remains to be done to bring the human engineering of computer-based systems to the same level of professionalism as hardware and software engineering (Gaines and Shaw, 1986a, b). For decision and control systems involve expert systems and simulation, information presentation and ease of communication are of prime importance (Gaines, 1986c). A technically powerful system will be wasted unless it is comprehensible and accessible to its users, and gives them confidence in the basis of its operations.

### Communication protocols

The days of single-vendor, special-purpose systems designed against a static, well-defined specification are long gone. Integrated manufacturing systems integrate all aspects of the operation of a firm, from its financial planning and accounting, through its inventory control, customer, order, delivery and payment records, production planning, scheduling, control and monitoring, to its liaison with other organisations concerned with distribution, marketing, environmental control, and so on. Equipment from many vendors, purchased at different times, for different purposes, by different parts of the organisation, has to operate together. In this situation lie the major practical obstacles to the creation of effective integrated manufacturing systems at present.

The technical problems are beginning to be resolved at the lower levels by the adoption of industry standards in communication protocols such as the Open Systems Interconnection (OSI) standard (Day and Zimmerman, 1983), the provision in data processing and database packages for the import and export of data from 'foreign' systems, and the provision of data communications for parameter setting and data collection in control

and instrumentation systems. These alone are necessary but not sufficient to allow the widespread creation of integrated manufacturing systems. They provide a syntactic framework for system integration but no coherent semantics in which to express it.

The Manufacturing Automation Protocol (MAP) has been developed to give application-level semantics to the OSI standard by General Motors (1984) as the major corporation most affected by CIM developments. It was sponsored by the GM Engineering and Manufacturing Computer Coordination activity in late 1979, placed in the public domain as a cooperative effort across computing, manufacturing equipment and manufacturing industries, and has attracted widespread industry support. MAP defines an application-layer protocol for OSI specific to integrated manufacturing system requirements. It is particularly significant in system design in offering the capability to collect operational and performance data on a scale, and with a richness of context, impracticable currently (McCarthy, 1985; Kaminski, 1986).

**The applications ring: Basic technologies**

The outer ring in Fig. 4.24 shows the main information technologies in integrated manufacturing systems and their typical applications. Moving anti-clockwise around the ring:

- *Remote communications* are necessary to interface the factory with its suppliers and consumers. A particular firm is part of an industrial infrastructure, each part of which is trying to optimise its productivity, profitability and stability. This can best be achieved if suppliers are aware of consumers' needs and projected needs, and consumers are aware of the production and delivery schedules of suppliers.
- *Computation systems* perform the classical number-crunching tasks under the control of algorithms that with modern software engineering have a high degree of accuracy, conformity to specification, and reliability.
- *Database systems* have become the classic core of a firm's information system, recording all its activities in a coordinated fashion and making the information available on a controlled basis to those systems and people with a legitimate need to know.
- *Control systems* began to be integrated with information systems as numerically controlled machine tools came into use, and with the development of industrial robots capable of part-handling, tool-changing and materials movement, total automatic of production has come into widespread use.
- *Instrumentation systems* have had to increase in capability to compensate for the fewer people observing the manufacturing process and the increased need for timely information on the state of the plant, tool

wear, and so on, to enable the integrated manufacturing system to optimise overall operation.

Three technologies in the outer ring of Fig. 4.24 are particularly relevant to decision aids for planning integrated manufacturing systems:

- *Knowledge bases* are distinguished from databases by containing inference rules rather than facts. The rules may necessarily be heuristic in nature, generating plausible advice rather than hard facts. If databases may be said to record a firm's activities, then knowledge bases may be said to record its skills, particularly its management skills.

- *Expert systems* are the means to apply a knowledge base to the solution of a particular problem. They stand in relation to the knowledge base as does conventional computation to the database. They differ in that their role as advisors and dependence on heuristics make accountability and comprehensibility of major importance. An expert system has to be able to justify and explain the basis of its advice.

  Expert system technology can be applied at a number of levels in CIM (Kusiak, 1986), including product design (Kim and Suh, 1985), FMS justification (Sullivan and LeClair, 1985), job-shop scheduling (Fox, 1985), simulation (Bruno et al., 1985), on-line process control (Brown et al., 1985) and robot control (Kirschbrown and Dorf, 1985). However, the problem of major concern is the design stage, at which the integrated manufacturing system has to be specified in terms of its performance and a range of diverse machine tools has to be brought together in an integrated system design. This stage currently presents a number of major conceptual difficulties that are impeding the rapid diffusion of integrated manufacturing system technology into appropriate industries. It is typical of the problems that because of their complexity and diffuseness they have to be 'managed' and are ideally suited to support through expert systems (Hayes-Roth, 1984).

- *Simulation* has been separated from other forms of computation because it is computation based on a model of some other system, and it is the relation of the simulated model results to that of the system modelled that is of paramount importance. The models used in simulation to infer the behaviour of a system are generally deeper than the models used in expert systems to infer the proper advice about a system, even though both may be concerned with the same system (Gaines and Shaw, 1985a; Shannon et al., 1985).

  Simulation technology ean be applied at a number of different levels in CIM (Smith, 1985). However, the planning stage for flexible manufacturing is one of the most significant (Wortman and Miner, 1985). An FMS involves the integration of a number of different tools, probably from different manufacturers and possibly in a way not

previously encountered by the manufacturers. There is a variety of semi-standard FMS configurations recommended or used to demonstrate possibilities, but the permutations required in practice go far beyond these. An FMS usually involves the integration of new technologies with older one. The newer technologies are particularly in material handling (Mangold, 1981; Edson, 1984) and in computer-based communication and software support for the overall system (Bruno et al., 1984). These are not well understood, and problems of matching flow rates between machines, and data rates and formats between equipment controllers, are severe. Simulation provides a manager with a way of experimenting with configurations and comint to understand the potentials of different tools (Gaines, 1986a; Winchmann, 1986).

The interactive decision-making process supported by expert systems and simulation is a natural way in which to manage FMS implementation. It would seem feasible to ascertain the requirements and then use non-interactive computation to provide an optimal solution, but requirements are rarely well defined and hard. Usually, a number of possibilities are open which are of diverse types, such as manufacturing at alternative sites or using different tools and sequencing. It is also often the case that the requirements are soft in that they will change as the simulation and decision process proceeds and the manager comes to understand the problem space better. These considerations are not so much defects of the human decision process as limitations of decision systems that do not have adequate sensitivity analysis. If a designer knew in using an algorithm for grouping tools that a particular constraint he had applied was reducing the throughput by 20%, then he might relax that constraint. However, it would place an excessive burden on the decision maker to require that all such trade-offs were specified in advance. The relevant 'what-ifs' constitute only a small subset of all possibilities. Interactive, sequential decision-making is an efficient method for determining such relevance. The combination of knowledge-based techniques and simulation is particularly attractive (Ben-Arieh, 1986; Shivnan and Browne, 1986). (See also Chap. 13 of this volume.)

## Summary

This section has presented the complex interrelations between a manufacturing organisation and the diverse information technologies necessary to implement CIM. Expert systems and knowledge bases are an extension of existing information technologies that are particularly important to management decision support, and integrate well with simulation. Communication is an important component of integration, but attention to the human-computer interface and to the effective organisational integration of the system is essential to success.

## Concluding remarks

Developments in computers, robotics, communications, electronics and materials are occurring at an increasing pace and have widespread impact on industrial manufacturing processes. In particular, the combination of these new technologies offers opportunities for integrated manufacturing systems that increase productivity through improved reliability, reproducibility, flexibility, human factors and cost-effectiveness. Integrated manufacturing systems are applicable to a wide spectrum of industry, from comparatively small firms up to large organisations. However, it is difficult for firms to keep up with the flow of information and experience on these rapidly changing new technologies.

Decision support systems for CIM present particular problems for today's expert system technology because manufacturing involves the dynamics of physical processes and requires knowledge-based systems that combine simulation techniques with those of current expert systems. They also present challenges in coordination because the expertise required is dispersed across a number of disciplines, between universities and industry, and between equipment manufacturers and users. However, the relevant technologies and knowledge are now at a stage where practical systems may be developed. The refinements of these systems, through experience in their use, further analysis and data collection, will lead to the evolution of the totally integrated manufacturing system.

## Acknowledgements

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada. The KITTEN implementation is a joint project with Dr. Mildred L. G. Shaw, and the PCS implementation is being carried out by the Alberta Research Council. The author is grateful to Mildred Shaw of the University of Calgary, John Boose and Jeff Bradshaw of the Boeing AI Center, and Ernie Chang of the Alberta Research Council for discussions relating to knowledge support systems. The author is also grateful to his colleagues at the University of Toronto, Harvey Kolodny, Neville Moray, Burhan Turksen and Tony Vannelli, for discussions relating to manufacturing systems.

## References

Adler, J. and Matheson, J., 1986. Expert knowledge elicitation using personal construct theory: Obtaining plant layout expertise with 'PLANET'. B.A.Sc. Thesis, Dept. of Industrial Engineering, University of Toronto.

Akeel, H. A., 1986. Role of robotics in flexible automation – An American perspective. In, *Proc. Japan-USA Symp. on Flexible Automation*, pp.3–4.

Bainbridge, L., 1979. Verbal reports as evidence of the process operator's knowledge. *International Journal of Man-Machine Studies*, 11(4): 411–436.

Bainbridge, L., 1986. Asking questions and accessing knowledge. *Future Computing Systems*, 1(2): 143–149.

Ben-Arieh, D., 1986. A knowledge-based system for simulation and control of FMS. In, Hurrion, R. D. (Ed.), *Simulation in Manufacturing*, pp. 287–295. IFS (Publications) Ltd, Bedford, UK.

Boose, J. H., 1984. Personal construct theory and the transfer of human expertise. In, *Proc. AAAI-84*, pp. 27–33. American Association for Artificial Intelligence, Menlo Park, California.

Boose, J. H., 1985. A knowledge acquisition program for expert systems based on personal construct psychology. *International Journal of Man-Machine Studies*, 23(5): 495–525.

Boose, J. H., 1986. Rapid acquisition and combination of knowledge from multiple experts in the same domain. *Future Computing Systems*, 1(2): 191–216.

Boose, J. H. and Bradshaw, J. M., 1986. Expertise transfer and complex problems using AQUINAS as a knowledge acquisition workbench for expert systems. In, *Proc. AAAI Workshop on Knowledge Acquisition for Knowledge-Based Systems*, pp. 5.0–5.16.

Bradshaw, J. M. and Boose, J. H., 1986. NeoETS. In, *Proc. North American Personal Construct Network 2nd Biennial Conf.*, pp. 27–41. Dept. of Computer Science, University of Calgary.

Broadbent, D. E., FitzGerald, P. and Broadbent, M. H. P., 1986. Implicit and explicit knowledge in the control of complex systems. *British Journal of Psychology*, 77: 33–50.

Brown, T., Alexander, S. M., Jagannathan, V. and Kirchner, R., 1985. Demonstration of an expert system for manufacturing process control. In, Birtwistle, G. (Ed.), *AI, Graphics and Simulation*, pp. 110–113. Society for Computer Simulation, La Jolla, California.

Bruno, G., Demartini, C. and Valenzano, A., 1984. Communication and programming issues in robotic manufacturing cells. In, *Int. Conf. on Robotics*, 84CH2008–1, pp. 361–367. IEEE Computer Society Press, Silver Spring, Maryland.

Bruno, G., Elia, A. and Laface, P., 1985. A rule-based system for production scheduling. In, *Proc. COMPINT '85*, IEEE 85CH2136–0, pp. 35–39.

Chang, E., 1986a. Participant systems: Group human-computer interaction. In, *Proc. 1986 Int. Conf. on Systems, Man and Cybernetics*, IEEE 86CH2364–8, pp. 1337–1342.

Chang, E., 1986b. Participant systems. *Future Computing Systems*, 1(3).

Collins, H. M., 1985. *Changing Order: Replication and Induction in Scientific Practice*. Sage Publications, London.

Davis, R., 1983. TEIRESIAS: Experiments in communicating with a knowledge-based system. In, Sime, M. E. and Coombs, M. J. (Eds.), *Designing for Human-Computer Communication*, pp. 87–137. Academic Press, London.

Davis, R. and Lenat, D. B., 1982. *Knowledge-Based Systems in Artificial Intelligence*. McGraw-Hill, New York.

Day, J. D. and Zimmerman, H., 1983. The OSI reference model. *Proc. IEEE*, 71(12): 1334–1340.

Deal, T. E. and Kennedy, A. A., 1982. *Corporate Cultures: The Rites and Rituals of Corporate Life*. Addison-Wesley, Reading, Massachusetts.

Dixon, N., 1981. *Preconscious Processing*. John Wiley & Sons, Chichester, UK.

Edson, D., 1984. Bin-picking robots punch in. *High Technology*, 4(6): 57–61.

Elavia, J. D., 1985. Where do I start with CIM? In, *Proc. Autofact '85*, pp. 1.1–1.21. Society of Manufacturing Engineers, Dearborn, Michigan.

Eshelman, L. and McDermott, J., 1986. MOLE: A knowledge acquisition tool that uses its head. Technical Report, Dept. of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania.

Feigenbaum, E. A., 1980. Knowledge engineering: The applied side of artificial intelligence. Report STAN-CS-80-812, Dept. of Computer Science, Stanford University, Stanford, California.

Fox, M. S., 1985. Knowledge representation for decision support. In, Methlie, L. B. and Sprague, R. H. (Eds.), *Knowledge Representation for Decision Support Systems*, pp. 3–26. North-Holland, Amsterdam.

Fransella, F. and Bannister, D., 1977. *A Manual for Repertory Grid Technique.* Academic Press, London.

Freud, S., 1914. *Psychopathology of Everyday Life.* Benn, London.

Gaines, B. R., 1977. System identification, approximation and complexity. *International Journal of General Systems*, 3: 145–174.

Gaines, B. R., 1986a. Expert systems and simulation in industrial applications. In, Luker, P. A. and Adelsberger, H. H. (Eds.), *Intelligent Simulation Environments*, pp. 144–149. Society for Computer Simulation, La Jolla, California.

Gaines, B. R, 1986b. Socio-economic foundations of knowledge science. In, *Proc. 1986 Int. Conf. on Systems, Man and Cybernetics*, IEEE 86CH2364-8, pp. 1035–1039.

Gaines, B. R., 1986c. Cognitive models of human-computer interaction in distributed systems. In, *Proc. 1986 Int. Conf. on Systems, Man and Cybernetics*, IEEE 86CH2364-8, pp. 1358–1362.

Gaines, B. R., 1986d. An overview of knowledge acquisition and transfer. In, *Proc. AAAI Workshop on Knowledge Acquisition for Knowledge-Based Systems*, pp. 14.0–14.18.

Gaines, B. R., 1986e. Expert systems and simulation in the design of an FMS advisory system. In, Hurrion, R. D. (Ed.), *Simulation in Manufacturing*, pp. 311–324. IFS (Publications) Ltd, Bedford, UK.

Gaines, B. R. and Andreae, J. H., 1966. A learning machine in the context of the general control problem. In, *Proc. 3rd Congr. International Federation for Automatic Control.* Butterworths, London.

Gaines, B. R. and Shaw, M. L. G., 1981a. A programme for the development of a systems methodology of knowledge and action. In, Reckmeyer, W. J. (Ed.), *General Systems Research and Design: Precursors and Futures*, pp. 255–264. Society for General Systems Research, Louisville, Kentucky.

Gaines, B. R. and Shaw, M. L. G., 1981b. New directions in the analysis and interactive elicitation of personal construct systems. In, Shaw, M. L. G., (Ed.), *Recent Advances in Personal Construct Technology*, pp. 147–182. Academic Press, London.

Gaines, B. R. and Shaw, M. L. G., 1984a. Dialog shell design. In, *Proc. INTERACT '84: 1st IFIP Conf. on Human-Computer Interaction*, Vol. 1, pp. 344–349. Elsevier, Amsterdam.

Gaines, B. R. and Shaw, M. L. G., 1984b. Logical foundations of expert systems. In, *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics*, pp. 238–247.

Gaines, B. R. and Shaw, M. L. G., 1984c. Expert systems: The substitution of knowledge for materials in world food production. *Possible Worlds*, 1(2): 5–8.

Gaines, B. R. and Shaw, M. L. G., 1985a. Expert systems and simulation. In, Birtwistle, G. (Ed.), *AI, Graphics and Simulation*, pp. 95–101. Society for Computer Simulation, La Jolla, California.

Gaines, B. R. and Shaw, M. L. G., 1985b. From fuzzy sets to expert systems. *Information Science*, 36(1–2): 5–16.

Gaines, B. R. and Shaw, M. L. G., 1985c. Foundations of expert systems. In, Kacprzyk, J. and Yager, R. R. (Eds.), *Management Decision Support Systems Using Fuzzy Sets and Possibility Theory*, pp. 3–24. ISR (Interdisciplinary Systems Research), Verlag TUV, Rheinland.

Gaines, B. R. and Shaw, M. L. G., 1985d. Systemic foundations for reasoning in expert systems. In, Gupta, M. M., Kandel, A., Bandler, W. and Kiszka, J. B. (Eds.), *Approximate Reasoning in Expert Systems*, pp. 271–281. North-Holland, Amsterdam.

Gaines, B. R. and Shaw, M. L. G., 1986a. From timesharing to the sixth generation: The development of human-computer interaction, Part I. *International Journal of Man-Machine Studies*, 24(1):1–27.

Gaines, B. R. and Shaw, M. L. G., 1986b. Foundations of dialog engineering: The development of human-computer interaction, Part II. *International Journal of Man-Machine Studies*, 24(2): 101–123.

Gaines, B. R. and Shaw, M. L. G., 1986c. Induction of inference rules for expert systems. *Fuzzy Sets and Systems*, 18(3): 315–328.

Gaines, B. R. and Shaw, M. L. G., 1986d. Knowledge engineering for an FMS advisory system. In, Lenz, J. E. (Ed.), *Proc. 2nd Int. Conf. on Simulation in Manufacturing*, pp.51–61. IFS (Publications) Ltd, Bedford, UK.

Gaines, B. R. and Shaw, M. L. G., 1986e. Knowledge engineering techniques. In, *Proc. AUTOFACT '86*, pp. 8.79–8.96. Society of Manufacturing Engineers, Dearborn, Michigan.

General Motors, 1984. General Motors Manufacturing Automation Protocol. General Motors Corporation, Warren, Michigan.

Gershwin, S. B., Hildebrant, R. R., Suri, R. and Mitter, S. K., 1986. A control perspective on recent trends in manufacturing systems. *IEEE Control Systems Magazine*, 6(2): 3–14.

Gevarter, W. B., 1983. Expert systems: Limited but powerful. *IEEE Spectrum*, 18: 39–45.

Goguen, J. A. and Meseguer, J., 1983. Programming with parametrized abstract objects in OBJ. In, *Theory and Practice of Programming Technology*. North-Holland, Amsterdam.

Gunn, T. G., 1981. *Computer Applications in Manufacturing*. Industrial Press, New York.

Gupta, M. M., Kandel, A., Bandler, W. and Kiszka, J. B., (Eds.), 1985. *Approximate Reasoning in Expert Systems*, pp. 271–281. North-Holland, Amsterdam.

Hall, E. T., 1959. *The Silent Language*. Doubleday, New York.

Hawkins, D., 1983. An analysis of expert thinking. *International Journal of Man-Machine Studies*, 18(1): 1–47.

Hayes-Roth, F., Waterman, D. A. and Lenat, D. B. (Eds.), 1983. *Building Expert Systems*. Addison-Wesley, Reading, Massachusetts.

Hayes-Roth, F., 1984. The industrialization of knowledge engineering. In, Reitman, W. (Ed.), *Artificial Intelligence Applications for Business*, pp. 159–177. Ablex, Norwood, New Jersey.

Hayes-Roth, B., 1985. A blackboard architecture for control. *Artificial Intelligence*, 26(3): 251–321.

Jaikumar, R., 1986. Postindustrial manufacturing. *Harvard Business Review*, 64(6): 69–76.

Khan, G., Nowlan, S. and McDermott, J., 1985. MORE: An intelligent knowledge acquisition tool. In, *Proc. 9th Int. Joint Conf. on Artificial Intelligence*, pp. 581–584. Morgan Kaufmann, California.

Kaminski, M. A., 1986. Protocols for communicating in the factory. *IEEE Spectrum*, 23(4): 56–67.

Kelly, G. A., 1955. *The Psychology of Personal Constructs*. Norton, New York.

Kempf, K. G., 1985. Manufacturing and artificial intelligence. *Robotics*, 1(1): 13–25.

Kernaghan, J. A. and Cooke, R. A., 1986. The contribution of the group process to successful project planning in R&D settings. *IEEE Transactions on Engineering Management*, EM-33(3): 134–140.

Kim, S. H. and Suh, N. P., 1985. On an expert system for design and manufacturing. In, *Proc. COMPINT '85*, IEEE 85CH2136–0, pp. 89–95.

Kirschbrown, R. H. and Dorf., R. C., 1985. KARMA – A knowledge-based robot manipulation system. *Robotics*, 1(1): 3–12.

Kusiak, A., Vannelli, A. and Kumar, K. R., 1985. Grouping problems in scheduling flexible manufacturing systems. Working Paper 84–16, Dept. of Industrial Engineering, University of Toronto.

Kusiak, A., 1986. Artificial intelligence and operations research in flexible manufacturing systems. Working Paper 10/86, Dept. of Mechanical and Industrial Engineering, University of Manitoba.

Landfield, A., 1976. A personal construct approach to suicidal behaviour. In, Slater, P. (Ed.), *Dimensions of Intrapersonal Space*, Vol. 1, pp. 93–107. John Wiley & Sons, London.

Lenat, D., Prakash, M. and Shepherd, M., 1986. CYC: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI Magazine*, 6(4): 65–85.

Liu, D., 1985. Intelligent manufacturing planning systems. *Autofact '85*, SME Technical Paper MS85–1070. Society of Manufacturing Engineers, Dearborn, Michigan.

Maimon, O., 1985. FMS real time operational control. In, *Proc. Autofact '85*, pp. 6.31–6.58. Society of Manufacturing Engineers, Dearborn, Michigan.

Majchrzak, A., Chang, T.-C., Barfield, W., Eberts, R. and Salvendy, G., 1987. *Human Aspects of Computer-Aided Design*. Taylor & Francis, Philadelphia, Pennsylvania.

Mangold, V., 1981. The industrial robot as a transfer device. *Robotics Age*, 3(4).

Mamdani, E. H. and Assilian, S., 1981. An experiment in linguistic synthesis with a fuzzy logic controller. In, Mamdani, E. H. and Gaines, B. R. (Eds.), *Fuzzy Reasoning and its Applications*, pp. 311–323. Academic Press, London.

Mamdani, E. H., Procyk, T. and Baaklini, N., 1976. Application of fuzzy logic to controller design based on linguistic protocol. In, Mamdani, E. H. and Gaines, B. R. (Eds.), *Discrete Systems and Fuzzy Reasoning*, pp. 125–149. Queen Mary College, London.

Marcus, S., McDermott, J. and Wang, T., 1985. Knowledge acquisition for constructive systems. In, *Proc. 9th Int. Joint Conf. on Artificial Intelligence*, pp. 637–639. Morgan Kaufmann, California.

McCarthy, J. J., 1985. MAP's impact on process plants. *Control Engineering*, 32(11): 67–69.

McDonald, M. E., 1985. The smart foreman – Expert systems for manufacturing. Autofact '85, SME Technical Paper MS85–1112. Society of Manufacturing Engineers, Dearborn, Michigan.

Michalski, R. S. and Carbonell, J. G. (Eds.), 1983. *Machine Learning: An Artificial Intelligence Approach.* Tioga, Palo Alto, California.

Michalski, R. S. and Chilausky, R. L., 1980. Knowledge acquisition by encoding expert rules versus computer induction from examples – A case study involving soyabean pathology. *International Journal of Man-Machine Studies*, 12: 63–87.

Michalski, R. S and Winston, P. H., 1985. Variable precision logic. *Artificial Intelligence*, 29(2): 121–146.

Michie, D. (Ed.), 1979. *Expert Systems in the Micro Electronic Age.* Edinburgh University Press, Edinburgh.

Miller, R. D., 1985. Simplification, management control, then intelligence automation. In, *Proc. Autofact '85,* pp. 1–14. Society of Manufacturing Engineers, Dearborn, Michigan.

Mitchell, C. M., Govindaraj, T. and Ammons, J. C., 1984. Human-machine interface in the control of flexible manufacturing systems. In, *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics*, 84CH2063–6, pp.59–62.

Morgan, C., 1984. *Robots: Planning and Implementation.* IFS (Publications) Ltd, Bedford, UK.

Nassr, J. J., 1985. Considerations in establishing a CIM environment. In, *Proc. COMPINT '85*, IEEE 85-CH2136–0, pp. 68–72.

Nisbett, R. E. and Wilson, T. D., 1977. Telling more than we can know: Verbal reports on mental processes. *Psychological Review,* 84:231–259.

Nygaard, K. and Dahl, O. J., 1981. The development of the SIMULA languages. In, Wexelblat, R. L. (Ed.), *History of Programming Languages*, pp. 439–480. Academic Press, New York.

Pearl, J., 1986. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3): 241–288.

Peray, K. E. and Waddell, J. J., 1972. *The Rotary Cement Kiln.* Chemical Publishing Co., New York.

Rappaport, 1986a. Task analysis in intelligent systems design. In, *WESTEX-86*, IEEE 86CH2332–5, pp. 61–67.

Rappaport, 1986b. Multiple problem spaces in the knowledge design process. In, *Proc. AAAI Workshop on Knowledge Acquisition for Knowledge-Based Systems*, pp. 37.0–37.15.

Rathmill, K., MaConail, P., O'Leary, S. and Browne, J. (Eds.), 1985. *Robotic Technology and Applications.* Springer, Berlin.

Reitman, W. (Ed.), 1984. *Artificial Intelligence Applications for Business.* Ablex, Norwood, New Jersey.

Rhodes, J. S., 1985. The development of integrated control systems for flexible manufacturing systems. In, *Proc. Autofact '85*, pp. 6.11–6.29. Society of Manufacturing Engineers, Dearborn, Michigan.

Rogers, C. R., 1967. *On Becoming a Person: A Therapist's View of Psychotherapy.* Constable, London.

Rosenblatt, F., 1958. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65: 386–407.

Roy, J., 1986. Expert systems in Nexpert. *MacTutor*, 2(2): 48–51.

Shannon, R. E., Mayer, R. and Adelsberger, H. H., 1985. Expert systems and simulation. *Simulation*, 44(6): 275–284.

Shaw, M. L. G., 1980. *On Becoming a Personal Scientist*. Academic Press, London.

Shaw, M. L. G., 1981. Conversational heuristics for eliciting shared understanding. In, Shaw, M. L. G. (Ed.), *Recent Advances in Personal Construct Technology*, pp. 31–44. Academic Press, London.

Shaw, M. L. G., 1982. PLANET: Some experience in creating an integrated system for repertory grid applications on a microcomputer. *International Journal of Man-Machine Studies*, 17(3): 345–360.

Shaw, M. L. G., 1986. PCS: A knowledge-based interactive system for group problem solving. In, *Proc. 1986 Int. Conf. on Systems, Man and Cybernetics*, IEEE 86CH2364–8, pp. 1353–1357.

Shaw, M. L. G. and Chang, E., 1986. A participant construct system. In, *Proc. North American Personal Construct Network 2nd Biennial Conf.*, pp. 131–140. Dept. of Computer Science, University of Calgary.

Shaw, M. L. G. and Gaines, B. R. 1980. Fuzzy semantics for personal construing. In, *Systems Science and Science*, pp. 146–154. Society for General Systems Research, Louisville, Kentucky.

Shaw, M. L. G. and Gaines, B. R., 1981a. The personal scientist in the community of science. In, Reckmeyer, W. J. (Ed.), *General Systems Research and Design: Precursors and Futures*, pp. 59–68. Society for General Systems Research, Louisville, Kentucky.

Shaw, M. L. G. and Gaines, B. R., 1981b. The personal computer and the personal scientist. *BSC '81: Information Technology for the Eighties*, pp. 235–252. Heyden Press for British Computer Society, London.

Shaw, M. L. G. and Gaines, B. R., 1981c. Exploring personal semantic space. In, Rieger, B. (Ed.), *Empirical Semantics II*, pp. 712–791. Studienverlag Brockmeyer, Bochum, West Germany.

Shaw, M. L. G. and Gaines, B. R., 1983a. A computer aid to knowledge engineering. In, *Proc. British Computer Society Conf. on Expert Systems*, pp. 263–271.

Shaw, M. L. G. and Gaines, B. R., 1983b. Eliciting the real problem. In, Wedde, H. (Ed.), *International Working Conf. on Model Realism*, pp. 100–111. Pergamon, Oxford.

Shaw, M. L. G. and Gaines, B. R., 1984. Deriving the constructs underlying decision. In, Zimmerman, H.-J., Gaines, B. R. and Zadeh, L. A. (Eds.), *Fuzzy Sets and Decision Analysis*, TIMS Series Studies in the Management Sciences. North-Holland, Amsterdam.

Shaw, M. L. G. and Gaines, B. R., 1985. Developments in dialog engineering. In, *Proc. National Computer Conf.* Vol. 54, pp. 493–499. AFIPS Press, Arlington, Virginia.

Shaw, M. L. G. and Gaines, B. R., 1986a. A framework for knowledge-based systems unifying expert systems and simulation. In, Luker, P. A. and Adelsberger, H. H. (Eds.), *Intelligent Simulation Environments*, pp. 38–43. Society for Computer Simulation, La Jolla, California.

Shaw, M. L. G. and Gaines, B. R., 1986b. Interactive elicitation of knowledge from experts. *Future Computing Systems*, 1(2): 151–190.

Shivnan, J. and Browne, J., 1986. AI-based simulation of advanced manufacturing systems. In, Hurrion, R. D. (Ed.), *Simulation in Manufacturing*, pp. 297–309. IFS (Publications) Ltd, Bedford, UK.

Shortliffe, E. H., 1976. *Computer-Based Medical Consultations: MYCIN*. Elsevier, New York.

Smith, C. A., 1985. The role of simulation analysis in improving the manageability of integrated systems. In, *Proc. Autofact '85*, pp. 11.1–11.10. Society of Manufacturing Engineers, Dearborn, Michigan.

Spur, G. and Mertins, K., 1985. Interactive production control in CAM. In, *Proc. COMPINT '85*, IEEE 85CH2136-0, pp. 572–576.

Su, Y. W., 1986. Modeling integrated manufacturing data with SAM. *IEEE Computer*, 19(1): 34–49.

Sullivan, W. A., 1985. Computer integrated manufacturing program justification. In, *Proc. Autofact '85*, pp. 2.34–2.55. Society of Manufacturing Engineers, Dearborn, Michigan.

Sullivan, W. G. and LeClair, S. R., 1985. Justification of flexible manufacturing systems using expert system technology. In, *Proc. Autofact '85*, pp. 7.1–7.13. Society of Manufacturing Engineers, Dearborn, Michigan.

Swartout, W. R., 1986. Knowledge needed for expert system explanation. *Future Computing Systems*, 1(2): 99–114.

Uenohara, M., 1986. Flexible automation in Japanese electronics industry. In, *Proc. Japan-USA Symp. on Flexible Automation*, pp. 5–12.

van de Brug, A., Bachant, J. and McDermott, J., 1985. Doing R1 with style. In, *Proc. 2nd Conf. on Artificial Intelligence Applications*, IEEE 85CH2215–2, pp. 244–249. IEEE Computer Society Press, Washington, DC.

Wahl, D., 1986. An application of declarative modeling to aircraft fault isolation and diagnosis. In, Luker, P. A. and Adelsberger, H. H. (Eds.), *Intelligent Simulation Environments*, pp. 25–28. Society for Computer Simulation, La Jolla, California.

Waterman, D. A., 1986. *A Guide to Expert Systems*. Addison-Wesley, Reading, Massachusetts.

Welbank, M., 1983. *A Review of Knowledge Acquisition Techniques for Expert Systems*. Martlesham Consultancy Services, BTRL, Ipswich, UK.

Wichmann, K. E., 1986. An intelligent simulation environment for the design and operation of FMS. In, Lenz, J. E. (Ed.), *Proc. 2nd Int. Conf. on Simulation in Manufacturing*, pp. 1–11. IFS (Publications) Ltd, Bedford, UK.

Wortman, D. B. and Miner, R. J., 1985. The role of simulation in designing manufacturing systems. In, *Proc. Autofact '85*, pp. 11.11–11.20. Society of Manufacturing Engineers, Dearborn, Michigan.