Intelligent System for Water Resources Licence Determination

Note Report

University of Surrey

R&D Note Record 406/1/S

NRA

National Rivers Authority

Project 406

Intelligent System for Water Resources Licence Determination

Khurshid Ahmad and Stephen M. Griffin

Research Contractors:
Artificial Intelligence Group,
Department of Mathematical and Computing Sciences,
University of Surrey,
Guildford,
Surrey GU2 5XH

National Rivers Authority,
30-34 Albert Embankment,
London SE1 7TL

R&D Note 406/1/S

Commissioning Organisation
National Rivers Authority
Rivers House
Waterside Drive
Aztec West
Bristol BS12 4UD

Tel: 0454 624400 Fax: 0454 624409

Dissemination Status
Internal: Released to Regions
External: Restricted

Statement of Use
This document may be used by the NRA for the appraisal of research proposals and identification of research priorities in the area of intelligent system development.

Research Contractors
This document was produced under R&D Contract 406 by :

Artificial Intelligence Group
Department of Mathematical
and Computing Sciences
University of Surrey
Guildford
Surrey GU2 5XH

Tel: 0483 300800
Fax: 0483 300803

NRA Project Leader
The NRA's Project Leader for R&D Contract 406 was :

Paul Shaw - NRA Southern Region

Additional Copies
This document is priced at £10 (inc. p&p). NRA staff wishing to purchase further copies should contact their Regional R&D Co-ordinator.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This report describes the development of an 'Expert Licensing System and Information Environment' (ELSIE). ELSIE was developed by the University of Surrey for the National Rivers Authority (NRA R&D Project 406) over a two year period between June 1992-June 1994. The principal objective of the project was to build an expert system for advising on matters related to water abstraction licensing. The ELSIE Project was undertaken following an earlier feasibility study conducted for the NRA by Wallingford Software and the University: the Water Resources Management Intelligent Assistant (W-RAISA, October 1990 - September 1991).

The ELSIE Project demonstrated how the knowledge of experts in water abstraction licensing can be collated, analysed and stored within a computer system for use as and when required by other community members. It used methods and techniques of knowledge engineering and of computer-based text analysis to build two information systems: an expert system and a legislation browser. An expert system is a computer program that mimics the behaviour of an expert in a narrow domain of knowledge. Legislation browser is a computer program that can help an abstraction licensing officer to access relevant aspects of the Water Resources Act 1991: the Act was indexed automatically and was 'marked-up' for use in an information retrieval program. The licence determination system has been implemented separately to the legislation browser system and the description of the two will be presented separately.

The expert system also contains a management monitoring system: an Application Progress Manager (APMan). This subsystem is essential because a licence is not granted instantaneously: the whole process can take up to three months as a number of public and private sector organisations may have to be consulted. ELSIE keeps a diary for the licensing officer and his or her manager.

The collation of the knowledge of an expert is by no means an easy task, particularly in highly specialised disciplines like abstraction licensing. Once the knowledge is collated it must be transformed such that an expert system can use it to infer new facts from pre-stored data and in making decisions. The collation process involves talking to experts, scripting the knowledge for the expert system and testing the system to check whether or not it behaves like the expert or not. This cycle of collate-analyse-script-test is labour intensive and time consuming. The University of Surrey was guided and encouraged in the collation by a group of abstraction licensing officers from four regions of the NRA, namely Southern, Thames, Yorkshire and Wessex regions. This group, the ELSIE Project Steering Group (PSG), met fourteen times over a two year period. They identified four specialised areas in abstraction licensing together with two experts in each area from NRA regions throughout the UK. Four experts were interviewed during a two month period (September - November 1992). The interviews involved asking the experts question designed to elicit the knowledge used by the expert in decision-making. The interviews were videotaped, transcribed and analysed by the University staff and the PSG. The abstraction licensing process was formalised in terms of a hierarchy of over 100 tasks organised in a four-deep

hierarchy together with 250 rules of thumbs used by the experts. The PSG was involved in specifying the contents of the expert system, in the design of ELSIE's user-interface, in testing the expert system at Surrey, and in setting up and evaluating the results of a mobile workshop for demonstrating ELSIE in three NRA Regions. Furthermore, the PSG liaised closely with the NRA-wide Abstraction Licensing Group which involved keeping the Group informed of project progress and demonstrating ELSIE at the 1993 Annual General Meeting of the Group.

The large 'volume' of knowledge collated in the ELSIE Project, over 34,000 words in the four video-tapes transcripts together with over 350,000 words of text collected by the University from the NRA, required the use of sophisticated knowledge acquisition techniques including a terminological analysis of texts, automatic rule-extraction from texts, brainstorming and structured walk-throughs of ELSIE's knowledge base. The PSG members spent the equivalent of 87 person days in the knowledge collation and system testing exercise. The University of Surrey human resources input was 120 person days.

The development of an expert system involves a trained knowledge engineer, a specialist computing scientist, whose task is to follow the collate-analyse-script-test cycle. The principal tool of the knowledge engineer is the so-called expert system development environment, a suite of programs that keeps track of and enables the engineer to execute the cycle as effectively as possible. The ELSIE project had additional demands of building a graphical user-interface (GUI), a database of applications and a numerical computation sub-system (for computing water balance). The ELSIE Project had to use a knowledge engineering environment with sophisticated GUI-building tools and a high-level programming language in conjunction with a database management system. ELSIE was written on a UNIX workstation in ProKAPPA™, which provides most of these facilities, and accesses the external proprietary databases ORACLE®. The effort spent in building the expert system by the University was about 100 person days and the effort spent in building other parts of the system was just under 100 person days. The adaptation of ELSIE as a management decision-making tool would require specialist computing support to maintain a system as diverse as ELSIE.

ELSIE's GUI is an emulation of the NRA's licence application form (WR-1-*) and other forms associated with the determination process (almost 20 forms from which over 80 dialog boxes have been designed). Most entries on the form have a menu that presents the user with a choice of input values predetermined by domain experts. Once the forms are filled, they are deposited in an ORACLE® database. The development of the GUI required a human resources input of 50 person days.

ELSIE was installed in three NRA Regions and was tested by various staff members during the period February 1994 - May 1994. These user trials exposed a number of shortcomings of the ELSIE expert system component: a total of 84 reported errors out of which 40 were 'fixed', ten 'errors' were really demands for expanding the knowledge base and nine 'errors' were reports of how slow the system was. Much was learnt by the PSG and the Contractors during user trials.

Note that the legislation browser performed much more robustly than the expert system component.

The various water resources Acts, Schedules to the Acts and the Statutory Instruments that help to execute the Acts are a crucial resource for the licensing officers whilst they determine abstraction licence applications. In order to satisfy this need to look up aspects of legislation, the University of Surrey analysed the problem, helped in procuring machine readable forms of the Water Resources Act 1991, and devised a system to automatically index the Act, and to retrieve the user determined Parts, Chapters, Sections or paragraphs of the Act.

The Legislation Browser was developed exclusively on a PC platform and the programs for the marking up and automatic indexing of the text were written using the built-in language of a proprietary word processing system (Microsoft Word™). The browser has been received enthusiastically by the licensing officers within the NRA and by other sections of the NRA.

The ELSIE system is a working demonstrator in that it successfully helps the user with a large proportion of the procedure required to determine a licence application. Moreover, a number of licensing officers thought that the system can be used as a valuable training tool for newly-inducted licensing officers. The scope of the system has been defined with some rigour, nevertheless since specialist knowledge is frequently pruned and modified, it would be important to update the system at regular intervals.

KEYWORDS

Abstraction, Artificial Intelligence, ELSIE, Expert System, Legislation, Licensing, Water Resources.

GLOSSARY

Antecedent: The lefthand side of a production rule. The pattern needed to make the rule applicable.
APMan: Applications Progress Manager, part of ELSIE, developed by the University of Surrey to aid NRA staff in licence application determination.
Artificial Intelligence (AI): A discipline devoted to developing and applying computational approaches to intelligent behaviour.
Artificial Intelligence (AI) Approach: An approach that has its emphasis on symbolic processes for representing and manipulating knowledge on a problem solving mode.
Backward Chaining: A form of reasoning starting with a goal and recursively chaining backwards to its antecedent goals or states by applying applicable operators until an appropriate earlier state is reached or the system backtracks. This is a form of depth-first search.
Cognition: An intellectual process by which knowledge is gained about perceptions of ideas.
Consequent: The right side of a production rule. The result of applying a procedure.
Data Base: An organized collection of data about some subject.
Data Base Management System: A computer system for the storage and retrieval of information about some domain.
Data-Driven: A forward reasoning, bottom-up problem solving approach.
Data-Structure: The form in which data are stored in a computer.
Default Value: A value to be used when the actual value is unknown.
Domain: The problem area of interest, e.g., bacterial infections, prospecting, VLSI design.
ELSIE: Expert Licensing System and Information Environment, developed by the University of Surrey to aid NRA staff in licence application determination.
Event-Driven: A forward-chaining problem-solving approach based on the current problem status.
Expert System: A computer program that uses knowledge and reasoning techniques to solve problems normally requiring the abilities of human experts.
Forward Chaining: Event-driven or data-driven reasoning.
Frame: A data structure for representing stereotyped objects or situations. A frame has slots to be filled for objects and relations appropriate to the situation.
Goal Driven: A problem-solving approach that works backward from the goal.
Heuristics: Rules of thumb or empirical knowledge used to help guide a problem solution.
Hierarchy: A system of things ranked one above the other.
Hypertext: A generic term used to cover a number of techniques to create and view multidimensional documents, which may be entered at many points and which may be browsed in any order by interactively choosing words or key phrases as search parameters for the next text image to be viewed
Infer: To derive by reasoning. To conclude or judge from the premises or evidence.
Inference: The process of reaching a conclusion based on an initial set of propositions, the truths of which are known or assumed.
Inference Engine: Another name given to the control structure of an AI problem solver in which the control is separate from the knowledge.
Intelligent Assistant: An AI computer program (usually an expert system) that aids a person in the performance of a task.

Interactive Environment: A computational system in which the user interacts (dialogues) with the system (in real time) during the process of developing or running a computer program.
Knowledge Base: AI databases that are not merely files of uniform content, but are collections of facts, inferences and procedures, corresponding to the types of information needed for problem solution.
Knowledge Base Management: Management of a knowledge base in terms of storing, accessing and reasoning with the knowledge.
Knowledge Acquisition: The use of various techniques to elicit and document an expert's knowledge for subsequent incorporation into an expert system.
Knowledge Engineering: The AI approach focusing on the use of knowledge (e.g., as in expert systems) to solve problems.
Knowledge Representation: The form of the data-structure used to organise the knowledge required for a problem.
Legislation Browser: A hypertext information retrieval system providing easy access to the entire contents of the 1991 Water Resources Act.
Object-Oriented Programming: A programming approach focused on objects which communicate by message passing. An object is considered to be a package of information and descriptions of procedures that can manipulate that information.
Pattern Matching: Matching patterns in a statement or image against patterns in a global data base, templates or models.
Premise: A first proposition on which subsequent reasoning rests.
Problem-Solving: A procedure using a control strategy to apply operators to a situation to try to achieve a goal.
Production Rule: A modular knowledge structure representing a single chunk of knowledge, usually in If-Then or Antecedent-Consequent form. Popular in Expert Systems.
Programming Environment: the total programming set-up that includes the interface, the languages, the editors and other programming tools.
Prototype: An initial model or system that is used as a base for constructing future models or systems.
SGML: Standard Generalized Markup Language: a family of ISO standards for labeling electronic versions of text.
Slot: An element in a frame representation to be filled with designated information about the particular situation.
Syntax: The order (grammar) of a language.
User Interface: The system by which the user interacts with the computer.
WALDES: Water Abstraction Licence Determination Expert System, part of ELSIE, developed by the University of Surrey to aid NRA staff in licence application determination.
W-RAISA: Water Resources Management Intelligent Assistant developed by the University of Surrey to aid NRA staff in licence application determination.

1. INTRODUCTION

1.1 Structure of this Report

This report comprises nine sections:

This introductory section provides the reader with a background of this report (1.2), including why the project was started (1.2.1) and an overview of the knowledge engineering process employed by the University of Surrey during the project (1.2.2). The background of the contractors who undertook the project for the NRA are detailed in Section 1.3 and the projects steering group is introduced in Section 1.4.

Section 2 contains a discussion of the project objectives as outlined in the project plan.

Section 3 describes the knowledge acquisition phase of the project. After an introduction to knowledge acquisition (3.1) we introduce the concepts of language, meaning and understanding (3.2), the theory on which we have based our knowledge acquisition methodology, described in Section 3.3. Details of expert interviews (3.4) are followed by a discussion of how the transcripts, or 'mature corpus', may be analysed to aid the knowledge acquisition process. Section 3.5 describes the use of other domain specific texts (or the 'initiation corpus') and Section 3.7 discusses the exploitation of all texts throughout the knowledge acquisition process, both before and after the interview. We summarise the knowledge acquisition process in Section 3.8.

Section 4 discusses the methods used in developing ELSIE, including a description of the components which make up ELSIE (4.1) and details of how the system was designed through the creation of a hierarchy of tasks which the system was to perform (4.2). Section 4.3 describes the implementation of ELSIE in an object-oriented and production rule environment.

Section 5 describes ELSIE as it appears to the user. Section 5.1 provides some general information on the use of ELSIE. Section 5.2 describes how to select an existing application from a database or start a new application before the two main ELSIE components are described in some detail: APMan in Section 5.3 and WALDES in Section 5.4.

Section 6 describes the Legislation Browser for the Water Resources Act 1991. Section 6.1 describes the 1991 Water Resources Act, followed by a discussion on how the Act relates to other Acts of parliament (6.2) and the importance of key words and phrases in the Act (6.3). Section 6.4 describes how hypertext documents, like the Legislation Browser, may be produced semi-automatically, and Section 6.5 describes how the Act can be accessed through the Legislation Browser.

Section 7 describes how ELSIE was tested in several regions of the NRA. It includes a summary of the bugs detected by the testers and the status of these bugs in the current version of the system.

Section 8 summarises the contents of this report.

Section 9 outlines recommendations (obtained from both the development team and NRA staff) for changes and enhancements to be made to the system if it is to enter an implementation phase to become a fully operational system.

1.2 Background to this Report

1.2.1 Background to the Project

R&D Project 241, Expert Systems for Water Resources Management was completed by the contractors in 1991 and successfully demonstrated the applicability of expert systems technology to the water resources function by developing a prototype expert system 'W-RAISA'. This prototype is able to provide guidance to water resource officers on abstraction licence determination and deals mainly with proposed groundwater abstractions for the purpose of spray irrigation.

During the development of W-RAISA it was quickly realised that the licence determination process was facilitated by the provision of a number of tools in addition to the expert system itself. These included a hypertext-like browsing facility for water resources legislation, look up tables for guidance on irrigation rates, and a spread sheet calculating facility to test for possible derogation of nearby interests. It was also appreciated that the system could provide an organisational facility for monitoring the progress of applications. This would help resource officers to keep track of current applications and enable them to respond rapidly to enquiries.

W-RAISA was reviewed at a meeting of water resources managers and has undergone trials by water resource officers from Southern, Thames, Yorkshire, Wessex and Anglian NRA regions. Its potential for further development has also been considered by the ALG.

Phase One of the expert system project was awarded after competitive tender to the Hydraulics Research Ltd (Wallingford Software) and University of Surrey. The subsequent creation of the W-RAISA prototype involved the use of knowledge acquisition methods developed specifically for the project by the University of Surrey.

Having completed Phase One, the Knowledge Engineer at the University of Surrey was familiar with the problem domain of abstraction licensing. Therefore Surrey University and Wallingford Software already have the relevant and unique expertise necessary to carry out Phase Two.

The contractors had gained experience in the use of a specific expert system programming shell (KAPPA) and established large data sets, from interviews and user trials, which were carried forward for use in phase two. The Knowledge Engineer at Surrey was also able to continue work on Phase Two.

This enabled the prototype expert system from the W-RAISA project to be fully developed to provide an intelligent assistant and data organiser for resource officers. Liaison was maintained with the ALG to ensure that requirements in a national context were fully addressed.

1.2.2 Knowledge Engineering: an Overview

ELSIE constitutes many computer science paradigms: artificial intelligence, object-orientated programming, human computer interaction, relational database management, hypertext systems, and so on. The development of the system also drew on many research paradigms outside the domain of computer science, particularly terminology, lexicography, group facilitation and consensus decision making.

Knowledge Engineering is the acquisition and subsequent implementation of an expert's knowledge into a computer system. The three main issues tackled during knowledge engineering, by the knowledge engineer, are knowledge acquisition, knowledge representation and knowledge dissemination.

Knowledge acquisition is the use of various techniques to elicit and document an expert's knowledge for subsequent incorporation into an expert system. Knowledge is often experiential, descriptive, qualitative and largely undocumented and its acquisition is generally regarded as one of the most difficult tasks associated with the development of expert systems. The strategy adopted for the prototype project is described in Ahmad et al. (1990).

Knowledge representation is the encoding of acquired knowledge into a computer system. Knowledge has been represented on computers in many ways, from the original production rules, through frames and semantic networks, to the currently popular object-oriented programming. At Surrey we use a hybrid of all of these systems, using object-orientation to model the domain in networks of frames, and production rules to model the problem-solving knowledge required to determine a licence application. To aid the knowledge engineer in this task, we use a proprietary software development environment called ProKAPPA™, which provides facilities for development in each of these knowledge engineering paradigms.

Dissemination of the knowledge stored in the system is the most important aspect as far as the end user is concerned: he or she must have access to the knowledge in a format easily understandable without prior experience of how the knowledge is encoded in the system. For this purpose, the knowledge engineer must develop sophisticated graphical user interfaces for ease of use and comprehension of the facts being presented. Also, the knowledge must be made available to a wider audience by storing the facts in a proprietary database. Again, to aid in this task, we use tools from the ProKAPPA™ development environment.

Our approach to the entire knowledge engineering process is illustrated in Figure 1 below.

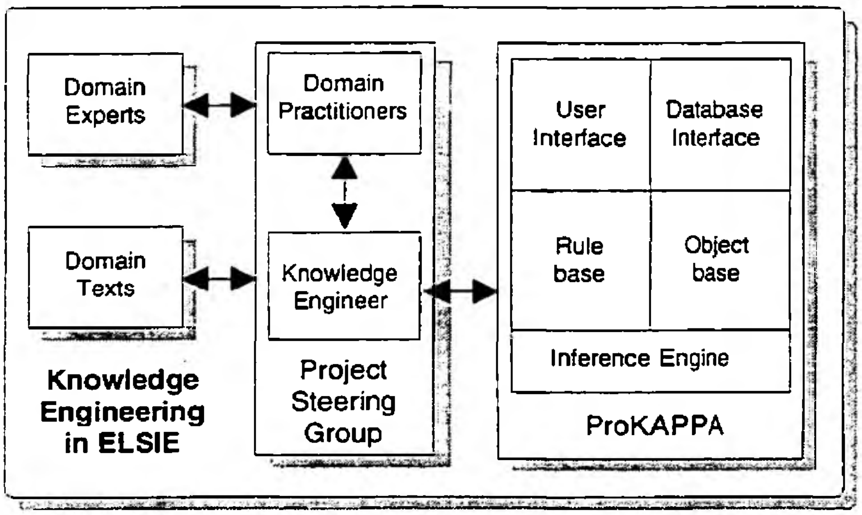


Figure 1: The knowledge engineering process

1.3 Background to the Contractors

1.3.1 Artificial Intelligence Group, University of Surrey

The Artificial Intelligence Group is part of the Department of Mathematical and Computing Sciences, University of Surrey. The group has extensive experience in knowledge acquisition and in structuring knowledge for expert systems, particularly applied to the water industry.

The ELSIE project team within the Artificial Intelligence Group consists of Dr. Khurshid Ahmad (Project Manager) and Stephen Griffin (Knowledge Engineer / System Developer). This is the same team that developed the W-RAISA prototype for R&D Project 241 of which this project is a continuation.

1.3.2 Wallingford Software

Wallingford Software specialises in providing solutions to all sectors of the water industry. These solutions comprise packaged software and associated training and support. Key product areas are urban drainage, river modelling, water treatment and coastal modelling. Wallingford Software is part of Hydraulics Research Wallingford.

Wallingford Software have advised on matters relating to the subsequent implementation of ELSIE. The ELSIE project member at Wallingford Software is the Technical Director Dr. Roland Price.

1.4 The Project Steering Group

Since the scope of the ELSIE system included all the main aspects of licensing on a national basis, it was necessary to seek the involvement of a larger number of experts than had been the case with the prototype system. A project steering group was formed to represent the views of the NRA nationally and to ensure that ELSIE received knowledge that was based on a national consensus. The group consisted of four experienced licensing officers drawn from four different regional offices of the NRA: Southern NRA, Thames NRA, Wessex NRA and Yorkshire NRA, together with the team at the University of Surrey. Paul Shaw, NRA Southern, acted as the project co-ordinator. One of the licensing officers provided a liaison with the NRA's national licensing working group: reporting to the group the progress of the ELSIE project and making the ELSIE project aware of the thinking of the group.

The project steering group held meetings where the overall scope of the ELSIE was discussed. This included a simultaneous consideration of the available knowledge and the requirements of a typical licensing officer.

2. OBJECTIVES OF THE RESEARCH

The overall objective of this project, as specified in the terms of reference at the start of the project, was to provide an intelligent assistant computer programme incorporating an expert system and a data organiser to aid abstraction licence application determination and monitoring by NRA water resources officers. This work would build on the existing W-RAISA expert system prototype developed in R&D Project 241 (see Ahmad and Griffin 1991 for details of the W-RAISA project).

Specific objectives were outlined as:

a) To provide a user-friendly WINDOWS interface to the components of the intelligent assistant computer programme which will comprise an expert system, a data organiser and a hypertext-like browser for legal data.

b) To broaden and deepen the scope of the W-RAISA prototype expert system such that the problem solving capability will include all the main categories of licence application as specified by the National Abstraction Licensing Group (ALG).

c) To provide an abstraction licence application data organiser.

d) To provide a hypertext-like browsing facility which will include the relevant legislation, policy guidelines and . example case histories and precedents.

e) To provide an industry standard data interface e.g. SQL (Structured Query Language) to data sources.

3. KNOWLEDGE ACQUISITION

3.1 Introduction

The use of psychological interviewing techniques in knowledge acquisition literature has been extensively documented. However, it is not clear how the output of these interviews, and indeed the preparation for these interviews, is undertaken systematically. This is particularly interesting in that the expert interviewee uses the so-called special language of his or her subject domain and the knowledge engineer, howsoever well-motivated, not being a member of the domain community does not have the necessary fluency in the special language of the expert. This special language, referred to, for example, as language of physics, language of microbiology, language of ancient arts, language of the military, language of meat production and so on, has its own idiosyncratic lexico-grammar.

The word 'lexico-' in the term lexico-grammar refers to the idiosyncratic choice of words, for instance repetition of key noun phrases in specialist language speech and text (*bacteria* and *virus* in microbiology; *atoms* and *nuclei* in nuclear physics, *frames and rules* in knowledge representation), the use of certain keywords/phrases to express taxonomies, the oft-repeated *ako* in AI literature, *a-part-of* to express part-whole relationships, *causes*, *produces,* and, *makes*, to express causality relations and a host of other lexical semantic relationships that may exist between domain objects and processes. It is rare to see references to literature on semantics, particularly terminology and lexical semantics.

Terminology being the science of how terms are coined, how terms enter the language of the specialist community, how it is refined and adapted linguistically and epistemologically, how the term and its variant are used, and how terms become obsolescent. Furthermore, terminologists work with computer scientists to specify, design and implement data bases, and increasingly simplistic knowledge bases, that contain terminology of a given domain.

Lexical semantics is the study of the meaning relationships between the lexical inventory of a natural language. Lexical semantics emphasises that word meaning can be dealt with exclusively in terms of relations between lexical items. Adequate accounts of word meaning must also take into account the fact that these relations should somehow be related to abstract concepts and the potential interrelationship between the concepts.

Both terminology and lexical semantics take a language- and use oriented view, or a special language view, of the terms of a specialist domain. And, whilst it is true that the needs of a terminologists and that of terminology users, for example translators, technical authors and so on, would be at some discernible variance with that of somebody involved in knowledge acquisition. There are lessons to be learnt by knowledge acquisition workers from terminologists (see for instance Picht and Draskau 1985, Jager 1990) and from lexical semanticists (see, for example, Cruse 1986, 1992).

Examples of the peculiar grammatical structures used by scientists, engineers and other specialists, include the preferential use of passives, nominalisation of verbs, etc. The lexico-grammatical idiosyncrasies in specialist speech and written texts evolve and persist due to the overriding need for unambiguous communication between the members of the domain community and between the community and people outside. The specialist domain community is particular about how the members use the specialist language of the domain. This care and attention manifests itself sometimes in the relative clarity of scientific documents, provided that one is familiar with the terms used in the domain and how the community expresses lexical semantic relationships using a restricted number of words and phrases.

Furthermore, the knowledge engineer spends a considerable amount of time animating the knowledge acquired from the experts. This is conducted in a *disjunctive* fashion: notes taken during the interview, or the interview transcript, serve as an *aide memoire* for the knowledge engineer. This aide memoire is interpreted subjectively by the engineer and programmed into a trial knowledge base through the use of a knowledge representation toolset. There are two points to note here. First, there is an enormous scope for personal bias in this interpretation in that if the same aide memoire is given to another knowledge engineer the results of the interpretation can be, in some cases, substantially at variance from each other. Second, the choice of toolset is crucial to the interpretation in that if the toolset has no empathy with the natural language constructs, the lexico-grammatical resources of specialist language, then the knowledge engineer has to translate twice: once from the experts language to their own and then from their own to the toolset language.

We would like to argue that a language-aware knowledge engineer, that is a knowledge engineer with an understanding of how terms evolve and are archived, and how terms can be interrelated at the linguistic level, would be able to avoid some of the knowledge acquisition bottlenecks quoted in the literature. These bottlenecks arise because the knowledge engineer is not familiar with the medium in which knowledge is communicated, that is the specialist language of the expert.

3.2 Language, Meaning and Understanding

Our research belief is that if knowledge engineers were able to exploit terminological, syntactic and semantic constructs used by experts for disseminating knowledge either through interviews or via domain texts, the knowledge acquisition process will not only be exploited there is a possibility that the exploitation of these constructs will also assure accuracy.

Before the interview [with the experts] the knowledge engineer, merely by collecting readily available domain text, can build and use a terminology collection of the domain for overcoming the inevitable terminological barrier between the knowledge engineer and the expert(s). Such terminology can be used in preparing a questionnaire for interviewing the expert and can also be used as a paper object-base of the putative knowledge. After the interview, textual analysis focused, for example, on extracting heuristics from the interview transcript (and other domain texts) or on extracting more terminology from the other domain texts for refining the object-base.

A knowledge engineer is expected to converse quite fluently with the experts of his or her target domain of application. Conversation, either for understanding the scope of the system or for eliciting problem-solving knowledge or debugging such knowledge, requires a modicum of understanding of the specialist language of the target domain. The knowledge engineer is expected to devise a set of substantially in-depth questions to ask of the expert such as to unravel knowledge which is generally the preserve of the expert. Once the knowledge is elicited and animated into a knowledge base the knowledge engineer is expected to explain what his or her system is doing in the language of the experts to novices, less well-endowed experts and peers of the expert. The knowledge engineer is expected to converse with different groups of people at different times during the course of building an expert system.

The expert is also used to expressing this knowledge at different levels and to different groups: for the novice in his or her domain through lectures or textbooks or technical manuals; for the lay-person through public lectures or popular science literature and public information literature including newspaper articles, notices and advertisements related to goods and services; and, last but not least, for the peers of the domain through specialist conference literature, including lectures and poster presentations, and through learned journals, chapters in edited collections of texts etc. The experts also have to communicate with their administrative line managers, for resources, for guidance, for reporting results of projects: this is usually expressed through telephone conversation, memoranda, letters etc. And, there are a range of texts that cannot be easily categorised, like specialist dictionaries, relevant entries in general-purpose encyclopaedias and various encyclopaedic publications like *handbooks*, *annual reviews*, and a host of other compendium.

In short, each specialism is underpinned by an archive of text ranging from the informative (for example, research papers and lectures) to instructive (for instance, text books, technical manuals) to the imaginative (like advertisements, letters of persuasion). The text and speech patterns range from the formal to the informal, and from the prepared to the spontaneous. In some text and speech patterns there is premium on maintaining cohesion and co-reference in texts, like text books and undergraduate lectures, whilst in others the assumption is that the reader is sufficiently experienced for the writer to not worry too much about occasional lapses of coherence and lack of co-reference. The domain archive is multidimensional. As we shall see later, it is the exploitation of this archive, parts of which are now readily available in machine readable form for most disciplines, that will prepare knowledge engineers for the systematic drawing up of interview questions, and it is the objective analysis of interview transcripts, in tandem with the domain archive, that enable the acquisition and debugging of problem-solving knowledge. Presently, we will continue with our attempt to highlight the importance of terminology as a resource and terminology as a science for making knowledge acquisition less ad hoc.

Members of any specialist domain are trained, almost from the day they are inducted into (or received into) the domain, into the whys and wherefores of looking up the meaning of specialist terms here, clarifying one long stretch of text by looking up an encyclopaedia there, discussing points of confusion with their peers and with the experts. Over the period of years, the members become aware of, and have more ready access to, the contents of the archive of the domain then say the information scientists and knowledge engineers.

But the apprenticeship of a knowledge engineer is neither as open-ended as that of the novices nor is the knowledge engineer motivated to become more involved in the domain matters beyond the calls of his or her duty of building an information system. Generally, knowledge engineer *reads up* about the domain, has some access to the domain experts. However in some exceptional expert systems projects has almost continuous access to the experts [1]. The knowledge engineer, in an ad hoc fashion, identifies key concepts of the domain, understands the meaning of salient terms, homes in on key stretches of texts, say in an interview transcript, for extracting heuristics.

The transcript is a good example of special language text: full of specialist terms and phrases; a narrative text that aims to inform its listener/reader. The knowledge engineer has to understand the terms, sentences, and long stretches of text to extract problem-solving knowledge and meta-knowledge that may prove useful for explanation and justification. Note the interview transcript is not a typical text of any specialist domain in that such text is not as frequently encountered in any domain as may be the case for the other informative, instructive and imaginative texts mentioned above.

Access to a terminology data bank should, in principle, alleviate problems related to the understanding of specialist terms. There are a number of complications in using a conventional terms bank. First, term banks are expensive to build and not every specialism comes ready with its own terminology data bank: in the case of emergent sub-disciplines of science and technology, a term bank is usually a post-dated artefact, available, if at all, after a gap of five to ten years. Second, assuming the term banks is available, the definition of a term, indeed definitions of words in a general language dictionary, are generally expressed in terms of between three to six other terms or words: the art or science of writing definitions is a fairly skilled task, is intertwined with open problems in philosophy and semantics, and, therefore in most term banks there are terms with pretty opaque and at time substantially circular definitions. Third, term banks are designed for the use of translators, documenters, and information scientists and consequently the cognitive bias in the design is more oriented towards language production and learning: it is, therefore, to be

---

[1]cf. MYCIN and DENDRAL literature shows how leading experts in micorobiology and chemistry repspectively worked in tandem with Stanford's Heuristic Programming Project: in the case of DENDRAL Buchannan et al published more than 10 research papers with leading experts in chemistry and MYCIN documentation acknowledges significant input from and involvement of the role of the staff at Stanford's Departments of Infectitious Diseases and Pharmacology. But both DENDRAL and MYCIN were ground breaking projects and were concieved in an world-environment where computers were sufficient novel to attract the continual attention of Nobel laureates and leading experts. This is certainly not the case now.

expected that data contained in the term bank will not enlighten a knowledge engineer about problem-solving tasks. And, fourth, the data structures used in the design of term banks stress the *atomicity* of individual terms: the use of relational tables, records and pointers, do not exploit the interconnectivity and interdependence of the terms of a specialism -- terms are described as atoms, capable of existing entirely on their own, an inert, structureless entity.

Despite the above mentioned reservations, we believe that whilst terminology data banks with their current structure and cognitive orientation may not be quite as relevant as the knowledge engineers would like to have, nevertheless, if there is an extant terminology data bank, the use of such a resource may cut down the expensive interaction with domain experts, an interaction that amounts sometimes to the expert jotting the definitions of terms. It is not just the data in the term banks that is crucial for knowledge acquisition, but recent innovations in exploiting text corpora for defining and elaborating general language words, specifically for constructing learners' dictionaries, and some forays in knowledge representation related to defining and asserting meaning of terms, are equally if not more important.

3.3 A Language-Aware Well Grounded Methodology

The question we ask in our knowledge acquisition research is on lines similar to that of lexicographers who build corpora, analyse the corpora, and use such corpora for finding words and for elaborating the meaning of words. A knowledge engineer can also build specialist corpora which can be used to list out potential single and compound terms of the domain, which can be used to extract problem-solving heuristics, which can be used for looking up the contextual examples of how a term is used, and which can be used to obtain explanatory and other meta-knowledge material.

We have used the plural term *corpora* to stress that there can be a number of corpora. For instance, there maybe one corpus for starting the dialogue with the domain community - the *initiation corpus-* and for designing key questionnaires for knowledge acquisition interviews, and another corpus for acquiring problem-solving knowledge - the *mature corpus*. The 'initiation' corpus may comprise a collection of texts, including excerpts from text-books, popular science articles and public information documents related to the target domain, newspaper texts and technical manuals: the initiation corpus is the source of the terminological evidence of the existence of terms, the source of potential rules and some explanatory material. The 'mature' corpus will contain expert's interview transcripts, learned papers written by the expert, and encyclopaedic material related to the domain: the maturity.

There are a number of well-developed methodologies in information retrieval, communications theory and corpus linguistics that are based simply on the frequency of occurrence of a linguistic token. Such frequency information can be used to design bandwidths of communication channels, for determining the choice of words used preferentially by a linguistic community, for the author attribution of literary texts and the forensic analysis of texts written by criminals and so on. More recently, the frequency based information has been used to extract terminological tokens in a specialist text by comparing the frequency of such tokens in these texts with a representative corpus of general language texts. Such a contrastive techniques can be, as we show below, applied to an analysis of the specialist corpora that can be assembled by a knowledge engineer as mentioned above.

Scientific and technical texts show a profusion of frequently occurring noun phrases: ranging from single nominal elements to adjective-noun combinations and right the way through complex noun phrases that have a prepositional element. The single nouns can be identified by their proportionate large frequency in specialist texts as compared to general language texts. The problem are the more complex noun phrases. There is some evidence that these complex noun phrases involve idiosyncratic use of punctuation, they are usually preceded by a small class of words, like determiners and so on: these idiosyncrasies result in typical patterns and these patterns can be easily detected.

Once an initiation corpus is assembled, usually a 100,000 word corpus is sufficient for knowledge acquisition purposes, then a terms list can be produced and passed onto domain experts for validation and verification. Once the terms list is approved, the knowledge engineer can query an extant term bank or enter the approved list in a specially adapted term bank. The corpus can be searched for the illustrative examples of the use of the terms list. This search will not only help the knowledge engineer in understanding and elaborating a term, but such data can be used for extracting explanation and justification data.

3.4 <u>Expert Interviews</u>

The geomorphological, agricultural and economic diversity within the UK means that the knowledge for ELSIE was acquired from experts from more than one region of the UK. Similarly, the system was successfully tested across the UK. Unlike other expert system projects, ELSIE had a steering group, comprising working licensing officers who in the normal course of their duties reported to the experts who were involved in much of the knowledge acquisition process.

3.4.1 Interviewing techniques

As the interviews of domain experts play a very important role in knowledge acquisition, so knowledge engineers have adapted interviewing techniques used by psychologists. Psychologists have developed these techniques for understanding how humans, both experts and novices, solve problems.

Various interview techniques are used to obtain different facets of the knowledge domain. For instance, some interviewing techniques help in the elicitation of high level problem solving strategies, whilst others help in the identification of the basic elements that describe the domain.

"Overview" interviews and "think-aloud" interviews are aimed at familiarising the knowledge engineer with the problem domain and they are likely to be held at an early stage in system development. These interviews also helped the knowledge engineer to determine the broad scope of the problem. "Structured Interviews" are expected to provide into how a given expert solves a problem, or more specifically, how the expert solves a particular facet of a problem.

The overview interview is conducted in the early phases of knowledge acquisition where the knowledge engineer and project steering group discuss the problems of the domain in broad terms on the basis of pre-prepared questionnaire. The think-aloud interviews are conducted during the course of the expert system development project for clarifying points of detail: here the knowledge engineer does not provide a questionnaire but lets the expert talk at length. In both these interviews the initiative rests with the expert and the knowledge engineer or the interviewer merely sets the scene and subsequently records the experts output. Structured interview, on the other hand, though based on a pre-prepared questionnaire, relies on the interviewer to set the experts some problems, and to seek clarifications about the domain problems with a view to elicit solutions and/or problem-solving strategies.

3.4.2 Brainstorming, consensus decision-making and debriefing

In order to co-ordinate the activities of the project steering group, we used techniques mentioned in the knowledge acquisition literature: brainstorming, consensus decision-making and debriefing (see, for instance Greenwell 1998 [4]). Brainstorming involved the knowledge engineer to briefly list out what he or she thinks are the key domain issues and then to invite a group of experts to comment. These issues can be presented, for example, as a set of "bullet points" derived from background literature of the domain. Alternatively, an interview can be presented for discussion: a video of the interview played back by the knowledge engineer whilst the experts comment on the content of the tape. This is followed by an annotation of the video transcript by the brainstorming experts: this can be regarded as an example of consensus decision-making. Once the topics or propositions are clearly identified and organised, say, in a network of interconnected topics, then the experts brainstorm again either to validate the network or to alter, or in some case redraw the network. The process of approving (or redrawing) the network involves extensive debriefing of the panel of experts by the knowledge engineer.

The ELSIE project used brainstorming (a) to select domain experts, (b) to script the interview questionnaire, (c) to analyse the interview, and (d) to confirm the validity of topics and propositions expressed by the expert. The questionnaire then formed the basis of a structured interview. Consensus decision-making was used in stages (c) and debriefing in stage (d).

3.4.3 Interview preparation

The steering group, through brainstorming sessions facilitated by the knowledge engineer, decided that ELSIE should target on four major functional problems faced by licensing officers within the UK:

(a) the legislative framework;
(b) licensing of groundwater sources;
(c) licensing of surface water sources; and
(d) impoundments.

The steering group, together with the project leader and knowledge engineer, decided that the best way to elicit knowledge of each of these key areas is to identify experts in each of the areas and to subsequently elicit knowledge about the area through a structured interview.

The expert's interview was to be pre-planned and pre-scripted in that the questions to be raised in the interview were discussed at length by the group and experts were encouraged to use visual material during the interviews and to review the overall content beforehand with the knowledge engineer.

3.4.4 Choice of experts

Interviews were conducted with domain experts for each of the four key licensing areas. The group chose the domain expert, the expert validator, and the interviewer from different NRA regions: the intention was to ensure as broad a coverage of knowledge and expertise as was possible. Table 1 shows the regional coverage by the ELSIE experts.

Interview Title	Domain Expert	Interviewer	Validators
Licensing of Groundwater Sources	Thames	Yorkshire	Anglia
Licensing of Surface Water Sources	Welsh	Wessex	Welsh / NW
Impoundments	Wessex	Southern	Thames
Legislation & licensing procedure.	Severn Trent	Thames	Anglian

Table 1: Regional coverage by ELSIE experts

One of the experts was expected to make himself available for a professionally-shot video-recorded interview and the interviewer was to be one of the licensing officers on the ELSIE steering group. The second expert was expected to see the video and make his or her comments known to the group. The first expert was the initial provider of knowledge, whilst the second expert was to verify the knowledge.

3.4.5 Validation of knowledge from the interviews

Once the interviews were shot the transcript of the interview was produced. The interview video and the transcript were sent to the two experts for comments. The domain experts were invited to edit the interview transcripts before they were further reviewed by experienced licensing staff from other NRA regions. In this way not only could the accuracy of the transcripts be checked but also the scope of knowledge coverage would be maximised within the limitations of the project. Furthermore the steering group, again through brainstorming sessions facilitated by the knowledge engineer, analysed the interview, and confirmed the validity of topics and propositions expressed by the expert.

The steering group deliberations helped the knowledge engineer to identify the key topics or propositions the experts use in determining the license applications. The steering group also used supplementary written material such as legislative texts, policy notes, operational manuals and case study reports in order to further focus the scope of the system.

3.5 The 'Mature Corpus' and its Initial Analysis

The four transcripts of the interview formed the mature corpus for the ELSIE project: the transcripts of the video, on average almost one hour long, amounted to almost 35,000 words. Table 2 shows the breakdown of each of the four interviews and also indicates how many 'rules' were extracted *semi-automatically* from each of the interviews.

Interview Title	**Duration**	**Transcript Length**	**No. of Topics or Propositions**	
	Minutes	No. of words	**Initial Count**	**Post Peer Review**
Licensing of Groundwater Sources	60	8202	69	64
Licensing of Surface Water Sources	60	10758	58	34
Impoundments	45	5993	56	32
Legislation & licensing procedure.	60	9714	142	65

Table 2: Interview Transcript and Analysis Details

We will be discussing the semi-automatic extraction of the rules in some detail in Section 3.5.4, for the time being we will be concentrating on the behaviours of specialist terms in one of the interviews (Item 1 in Table 2).

Figure 2 is an excerpt of the first interview.

There are over 8000 words in this transcript, some like *the*, *of*, and *to* are used very frequently: counting the frequency of occurrence of these lexical tokens in the transcript one finds 521 occurrences of the token *the* (6.4% of the text), 295 and 289 occurrences of *of* and *to* (3.6% and 3.5 % of the text respectively). These three most frequently occurring words therefore comprise over 13% of the text.

When we are making decisions about the development of ground water resources, granting licences, it is very important to realise, basically, how aquifers behave and to realise that there is quite a range in the behaviour of aquifers.

If we look at how they occur in relation to geological structure we can distinguish between two basic types - confined aquifers, where the water is held under pressure by a layer of impermeable strata above the aquifer, and unconfined aquifers in which there is a free surface water table which is exposed to the atmosphere. These two different conditions of being confined and unconfined have quite a fundamental effect on how water level in the aquifer responds to pumping and that is the basic process that we have to consider in making our decisions. With a confined aquifer, the aquifer is under pressure and the removal of a certain quantity of water causes a much bigger effect on ground water level than is the case with an unconfined aquifer in which the loss of water from storage is not a pressure effect but it is gravity-drainage.

Figure 2 Excerpt of Interview Transcript for ELSIE Project

Now if we look at nouns in the text, including singulars and plurals, the term *water* makes up only 0.8% of the text (frequency 73 out of 8202) and the 14 most frequently occurring terms - water, aquifer, test, abstraction, river, consent, licence, groundwater, borehole, resource, environmental, pumping, flow and activity - comprise only 6% of the text. Note that these domain specific terms, and as we show very frequently used terms, still comprise less of the 8202 word text than the determiner *the*.

	Word /	Absolute Frequency			Percentage
	Term	Sing.	Plur.	Total	Of Text
Most	the	521		521	6.35
Frequent	of	295		295	3.60
Words	to	289		289	3.52
	water	68	5	73	0.89
Most	aquifer	37	27	64	0.78
Frequent	test	45	6	51	0.62
Domain	abstraction	36	7	43	0.52
Specific	river	34	5	39	0.48
Terms	consent	25	5	30	0.37
	licence	19	10	29	0.35

Figure 3: Occurrence of common words and domain specific terms in an interview transcript

The preponderance of determiners, modal verbs, prepositions etc., classified as the so-called closed class words in that it is only over centuries that new words are either added or subtracted from this class (cf. *thee* and *thou* are excluded in English) is a curious statistic of written and spoken language, something like 200 words make up 50% of the words used by speakers and writers of any language. The rest of the stock. millions of tokens, make up for the other half of language in use. These are called open class words, precisely nouns, adjectives and full verbs, in that new words are constantly being added to this category.

The distribution of word classes in the above transcript is very different to the distribution of word classes used in the English of everyday usage. This variance from the language of everyday - general language - is often referred to as the 'weirdness of special language' as we show below.

3.5.1 Weirdness of special language and representative corpora of general language texts

If we look at the frequencies of terms such as *water*, *river*, *abstraction* and *abstractions*, *catchment*, *borehole* and *groundwater*, we find that the relative frequency of these terms is an order of magnitude greater than the relative frequency of these words in a representative corpus of general language: the Longman Corpus of Contemporary English (see Table 3).

Word	Freq.	Transcript Rel. Freq. (SL)	Longman Rel. Freq. (GL)	Ratio
aquifer	37	4.51E-03	0	INF
drawdown	7	8.53E-04	0	INF
groundwater	29	3.54E-03	3.88E-07	9104.42
borehole	22	2.68E-03	4.85E-07	5525.44
catchment	8	9.75E-04	3.88E-07	2511.56
abstraction	36	4.39E-03	5.92E-06	741.12
licence	19	2.32E-03	5.53E-06	418.59
abstractions	7	8.53E-04	3.79E-06	225.40
river	34	4.15E-03	1.29E-04	32.20
water	68	8.29E-03	4.77E-04	17.38

Table 3: Relative frequency of terms in special language and general language and their ratios

Compare the ratio for these open class words to some of the closed ones in Table 4.

Word	Freq.	Transcript Rel. Freq. (SL)	Longman Rel. Freq. (GL)	Ratio
the	521	6.35	6.09	1.04
and	220	2.68	2.80	0.96
I	47	0.57	1.08	0.53

Table 4: Relative frequency of closed class words in special language and general language and their ratios

When the ratio of SL/GL for a word is greater than 1, then this word is more frequently used in special language than the equivalent use in general language. However, if this ratio is less than 1, then the opposite situation prevails.

It is possible to use the SL/GL ratio for identifying a term in a text and this analysis would, as we show, be very productive on the initiation corpus in that *before* the interview, the knowledge engineer can have a list of potential terms in his/her possession.

A more detailed description of term identification through the weirdness of special language, and details of representative corpora of general language texts, can be found in R&D Project Record 406/3/S.

3.5.2 Concording texts

A concordance of a text provides an index of all words in a text corpus showing every contextual occurrence of a word. If we were to make a concordance of the interview transcript and focus on the first two *nouns*, that is *abstraction* and *aquifer*, we find the examples in Figure 4 below.

1_154 ... making the decision as to whether or not an **abstraction** proposal is acceptable ...
1_116 There will shortly be starting a national research project aimed at collecting information on the main **aquifer** properties of transmissivity and storativity for a wide range of **aquifers**
1_32 No, there are no restrictions on volume as far as private domestic **abstractions** are concerned, they are always very small anyway, but if they should exceed 20 cubic metres a day, which is very unusual from groundwater, then they do become licensable ...
1_22 **Aquifers** may either have a granular matrix or fissured matrix and these two characteristics do have quite a marked effect, particularly on the proportion of the **aquifer** that is available to store water - a granular **aquifer** has a much larger proportion of volume which is available for storage of water, and this can be seen in the way that **aquifers** respond to changes in water level ...
1_196 I think prescribed levels are perhaps a bit more of a novelty but they are more seriously being considered now but prescribed flows, yes, where you know that the groundwater **abstraction** is going to affect river flow...
1_18 ... we can distinguish between two basic types - confined **aquifers**, where the water is held under pressure by a layer of impermeable strata above the **aquifer**, and unconfined **aquifers** ...

Figure 4: Extracts from a concordance produced by System Quirk from an interview transcript

Every contextual example of *aquifer* (and *abstraction*) throws some light on the meaning of the word and how it is used. Typical of any frequent domain *noun*, each of these is used as a carrier word: seldom used on its own and usually used in conjunction with an adjective (+ noun) combination. Furthermore, experts do actually define terms, mainly compound terms, as they conduct discourse. They use hyponymies (cf. *aquifer types*: confined aquifer and unconfined aquifer), attributes (aquifer properties: transmissivity and storativity), causal (aquifer responds to change in water level), qualifications (granular aquifer has a much larger proportion of volume...) etc. (see Table 5 below).

Object	Property	Possible Values
abstraction	proposal	acceptable / unacceptable
	type	private domestic
	volume	number
	source	groundwater
	licensable	yes / no
aquifer	type	unconfined / confined
	properties	aquifer properties object
	water level	number
	matrix	fissured / granular
	prescribed level	number
aquifer properties	storativity	number
	transmissivity	number
river	flow	number
	prescribed flow	number

Table 5: Possible object-property-value tuples elicited from the output of Figure 4

3.5.3 Compound terms and their distribution

Compound terms are the mainstays of any specialist text. General language texts seldom use compound words with the same frequency as special language texts. Compound terms can be identified by the assumption that they must make up any text falling between two so-called *closed-class* words, e.g. determiners, auxiliary verbs, conjunctions and so on. Figure 5 illustrates some compound terms in the interview transcript.

Compound terms usually provide the knowledge engineer with more object-base structure than the single word terms. Firstly, compound terms are often directly convertible into object-property pairs. From Figure 5, for example, the knowledge engineer may elicit the fact that an object *aquifer* will have the simple property *area*, and complex properties *parameters* and *properties*, both of which will probably reference other objects.

Candidate Term	Candidate Term	Candidate Term
abstract water	aquifer area	groundwater abstraction
contaminated waters	aquifer parameters	groundwater abstraction licences
fossil waters	aquifer properties	groundwater areas
intercept water	confined aquifer	groundwater catchment
irrigation water	fissured aquifer	groundwater catchment boundaries
polluted waters	granular aquifer	groundwater hydraulics
saline water	intergranular aquifer	groundwater licence
surface water	unconfined aquifer	groundwater licences
water company		groundwater protection
water resources	abstraction proposal	groundwater regimes
water wells	authorised abstraction	groundwater resources
	domestic abstraction	groundwater typology
environmental duties	irrigation abstraction	
environmental problems	licensed abstraction	
environmental value	net abstraction	

Figure 5: Compound terms in an interview transcript

Secondly, the compound may provide possible values for an object-property pair. Again, from Figure 5, the knowledge engineer may decide that the *aquifer* object must have a property say *type* with the value *confined*, *unconfined*, *fissured*, *granular* or *inter-granular*. Further values are often discovered by looking at the compound term in context, through concordancing as described above.

Finally, compound terms may hint at a possible hierarchy of objects. Figure 5, for example, hints that an object *water* could have a property *type* with possible values *contaminated*, *fossil*, *intercept*, *irrigation* etc, or perhaps *water* could be a superclass object with subordinate objects of *contaminated water*, *fossil water*, *intercept water*, *irrigation water* etc in a *is-a* or *type-of* hierarchy, and, as mentioned above, it is likely that objects of *aquifer parameters* and *aquifer properties* will be required to be referenced by the *aquifer* object.

One can concord the use of the above candidate terms and extract from the concordance information about the (lexical) semantic relations of the compound terms with other terms (cf. section 3.5.2).

3.5.4 Extracting Rules

The typical problem solving rules are written as IF X THEN Y or IF X AND Y THEN Z or IF X THEN Y AND Z. Indeed, if one were to search for the above pattern in the interview transcript one would find the following:

IF	casing is required in the borehole
THEN	careful geological control is needed
AND	it may be necessary to carry out geophysical borehole logging.

However, there are a whole range of words and phrases used by experts to encode heuristics of the types mentioned above (see Table 6 below).

affect	as a rule	as long as	assuming
because	customarily	due to	effect of
generally	hypothesis	if	if then
in general	ordinarily	precondition	premise
provided	proviso	reason	regularly
rule of thumb	seldom	so that	to ensure
typically	unless	usually	when

Table 6: Examples of semantic cues for locating 'rules' in text

Now, if we were to look at the interview transcript with cue words *reason*, *if* and *then*, then we may determine the rule shown in Table 7 below. This is the simplest type of rule but more complex types may also be discovered with this method (see Section 3.7.3).

Semantic Cues	Sentence in Transcript	Paper Knowledge Base Rule
	There is an exception in the case of Water	**If** Water Companies use pipes that
	Companies who make discharges of tests	have a diameter of more than 227mm
	through very large diameter pipes for	**then** a Discharge Consent is needed.
reason	some strange **reason**, the legislation has	
	picked on the size of 227 mm diameter	
if ..	which is the metric equivalent of nine	
	inches, so **if** they use a pipe which is more	
then	than that diameter **then** we are in the	
	rather strange position of a Consent being	
	needed under those circumstances.	

Table 7: Example of candidate simple rule found with semantic cues

3.6 <u>The 'Initiation Corpus'</u>

The knowledge acquisition session was preceded by organising a collection of texts emanating from the water resources licensing and abstraction domain. The text types collected here were quite varied. The domain experts were asked to provide documents related to their everyday work. These included documents written by others and official documentation like forms etc. Our experts, in general, were geologists, geographers or water engineers, and the area of law was one in which they used documentation extensively. Table 8 below describes the initiation corpus.

Text Type	Title	Audience	Length
Learned Book	Wisdom's Law of Watercourses	Lawyers / Water resources operatives	223960 words
Legislation Text	Water Resources Act 1991	Lawyers / Water resources operatives	127977 words
Legislation Text	Statutory Instruments	Lawyers / Water resources operatives	843 words
Official Documentation	Abstraction / Impoundment Licence Technical Officer's Report Abstraction Licence Impoundment Licence	Water resources operatives / General public	8793 words

Table 8: The initiation corpus

These documents are on the whole in machine readable form, and one of them, the Water Resources Act 1991, is available as a marked-up SGML document. Legal documents, generally impenetrable to non-lawyers, are an excellent source of definitions of terms. Although such definitions are sometimes controversial, and many a law suites are fought to change such definitions, they are perhaps the only source of definitions written by articulate people in conjunction with experts.

The initiation corpus contains documents which may not be regarded as text in the understood sense of the word. For instance, is a *form* a kind of a text? The answer is a possible yes and a possible no. Forms do not have this fundamental quality of texts, that is, the maintenance of cohesive order in the text by repeating and paraphrasing key lexical items.

This corpus contains a majority of terms used by the experts in their interviews. A foreknowledge of these terms, not actually used by us because of the logistics of obtaining the above texts, would certainly have prepared us much, much better for the interview.

Note that two of the texts in our initiation corpus are legal texts. Given that there can be three different types of rules within a legal text - definition rules, 'action' rules and stipulation rules (Bhatia 1993), it is no surprise that these texts contain definitions of engineering artefacts. The advantage of these definitions is that they are recognised by the English Law Courts unless given evidence to the contrary, and that the legal draftsmen do write exhaustive definitions. For example, a search for patterns like *X is a Y* revealed the following definitions that extend the common-sense meaning of concepts (taken from Wisdom's Law of Watercourses):

> A reservoir is a "raised reservoir" if it is designed to hold, or capable of holding, water above natural level of any part of the land adjoining the reservoir.

A public ferry is a public highway of a special kind.

We discuss how these definitions can be used in building knowledge bases in section 3.7.

3.7 Exploiting Texts Before and After

In the previous section we showed by a variety of examples how the textual analysis of terminological, syntactic and (lexical) semantic constructs of texts that comprise the mature and the initiation corpora can help in the identification of complex domain objects, in the identification of heuristics and so on.

The statistical tasks of computing the frequency of words, the recognition of patterns like *'X is a Y'* can, in principle, be carried out manually. However, not only will this process be very time-consuming, further bottlenecking the knowledge acquisition process, what is simultaneously required is the management of corpora and of terminology databanks. Furthermore, one needs access to a database of frequencies of words used in general language, and access to words and phrases that stand in for lexical semantic and causal relations.

3.7.1 The exploitation of the quirkiness of scientific texts

For the management of text corpora and termbanks, we have used System Quirk - a toolkit dedicated to the text-based extraction of terminology and of problem-solving knowledge. System Quirk[2] is a tool kit to aid in the analysis of text and the development of lexical and terminological database resources. The system can be used for a variety of purposes, including the development of term banks, the construction of dictionaries or the engineering of knowledge bases. The toolkit enables terms, words or domain objects to be acquired, elaborated, represented and disseminated from a corpus of texts. It comprises tools for creating, examining and extracting relevant material from evidence sources such as an organised special language text corpus, and tools for creating, deleting, modifying and maintaining a reference source such as terms in a terminology data bank: it covers the entire life-cycle of terminology management. Figure 6 shows the toolbox.

[2] System Quirk was developed by Paul Holmes-Higgin, Stephen Hook, Stephen Griffin and Syed Sibte Raza Abidi, University of Surrey, for the Translator's Workbench Project.

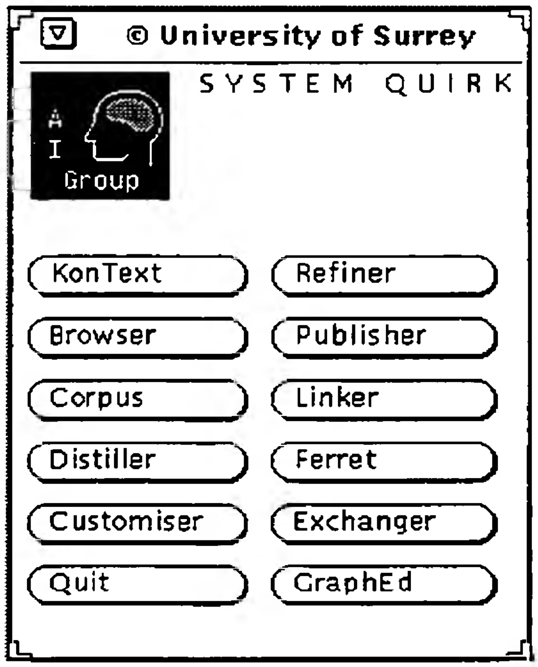


Figure 6: System Quirk toolbox

Table 9 below shows the functional characteristics of the tools.

Analysis Tools	Organisational Tools	Explicational Tools
Text Analysis: Concordance, Collocation, Statistical Analysis. KONTEXT	*Corpus Organisation:* Classification and Representation of full text units (books, papers, pamphlets, legal documents) according to information science principles. Organisation along pragmatic lines. CORPUS MANAGER	*Selective Explication:* Access within and across corpus, goal-oriented browsing. KONTEXT
Lexica/Term Analysis: Relationships with other lexical items, foreign language equivalents, etc. LINKER	*LDB/Term Bank Organisation:* Mapping of term bank fields onto proprietary database management systems. Creation and maintenance of data banks - including quality control measures. EXCHANGER DISTILLER	*Illustrative Explication:* Selection of illustrative text fragments - contextual examples; matching data in data base; representing relations. KONTEXT REFINER BROWSER GRAPHED PUBLISHER FERRET

Table 9: Functional characteristics of the toolsets in System Quirk

System Quirk is interfaced to a number of termbanks and to a database of frequency of general language words, computed from the Longman Contemporary Corpus of British English.

3.7.2 Extracting terms or domain objects and properties

The most novel feature of System Quirk is its ability to compare relative frequency of occurrences of words in a specialist corpus with that of the frequency of identical words in a 'representative' general language corpus like the Lancaster-Oslo/Bergen Corpus of British English (Hofland and Johansson 1982) and the Longman Corpus of Contemporary English prepared by the lexicographers and linguists of the dictionary publishers Longmans in the UK.

Words with a high ratio of occurrences in the specific text to the general text (which are marked as such by the system) are considered potential or 'candidate' terms of the specialist domain, as described in Section 3.5.1.

3.7.3 Extracting rules from text

System Quirk has facilities for identifying collocations. The collocation identifier functions are based on a key words in context (KWIC) type of analysis. The identifier extracts clauses and sentences which may be construed as the so-called heuristic rules of the domain, or as high-level problem-solving tasks, or as descriptions of 'objects' of the domain.

For instance, the rules are typically expressed as *IF<condition> THEN <action>* clauses or sentences. Collocation patterns containing *IF* and *THEN* can be searched for in the corpus, and on the basis of this search the knowledge engineer can present the domain expert with rule 'candidates'.

In order to facilitate the identification of the rules, a 'glossary' of words and phrases that are used to encode the rules in text has been organised. These words and phrases, or lexical semantic cues, have been extracted from Quirk et al (1985) and from various thesauri of English. A sample of the rule indicators compiled at the University of Surrey was presented in Table 6: Examples of semantic cues for locating 'rules' in text, in Section 3.5.4.

Given the interview transcript in a machine-readable form and some limited training in the use of the system, a linguist used System Quirk to analyse the text. The results were surprising. The linguist found 58 rules of which ten were amended and only three deleted by a knowledge engineer, who subsequently added only 19 more rules after a manual examination of the transcript.

Working with full sentences of natural language enables the retrieval of different levels of production rules. First, we have the simple rule: *if a then b*, an example of which was shown in Table 7: Example of candidate simple rule found with semantic cues, also in Section 3.5.4.

More complex rule types may also be discovered with this method. Some rules contain more than one antecedent and / or consequent, such as *if a and/or b then c and d*, as in the example in Table 10 below.

because **when** **so that**	Effluent returns, as I mentioned there, that's a very important thing **because** it can make the difference between an acceptable or non-acceptable abstraction, particularly **when** one is looking to see the effluent results, particularly, say from water supply abstractions, is coming back to the system above the point of abstraction, or very close to it, **so that** you can be utilising a process of re-circulation of the resource.	**If** the effluent is coming back to the resource system above or very close to the point of abstraction **then** a process of re-circulation of the resource can be utilised **and** a non-acceptable abstraction may become acceptable.

Table 10: Example of candidate complex rules found with semantic cues

There are also qualified rules: by which we mean rules with an attached excerpt of text that either extends the scope of the rules or censors the scope of the rule, for example default values for use in the rules, formulae for calculating values in the rules, or exceptions to the rule, illustrated in Table 11 below.

if	These two components - interception and induced recharge - add up to give the total depletion of river flow and the amount of depletion develops with time and with aquifer properties, but ultimately **if** you've got continuous abstraction in an open unconfined aquifer, you will, at some point, reach the situation where you've got 100% of the groundwater abstraction at the expense of the riverflow.	Interception and induced recharge add up to give the total depletion of river. The amount of depletion varies with time and aquifer properties. **If** there is continuous abstraction in an open unconfined aquifer **then** there will be 100% of the groundwater abstraction at the expense of the river flow.

Table 11: Example of candidate rules with qualifying propositions

Further, we have rules with explanations. These explanatory excerpts often provide further qualification of a rule but they are particularly useful for passing information back to a user who wants to know why a rule was fired, why a particular conclusion was drawn or why a decision was made.

An example of such an explanation is shown in Table 12 below.

affect **effect ..** **of**	... where you know that the groundwater abstraction is going to **affect** river flow, then the unacceptable environmental **effect of** that is to deplete river flow so there will be situations where you have got to limit the use of the abstraction by relating it to a prescribed flow ...	**If** a groundwater abstraction is going to affect the river flow **then** the abstraction should be limited by a prescribed flow **because** depletion of the river flow is an unacceptable environmental effect.

Table 12: Example of candidate rules with explanations

Finally, a single paragraph in the natural language may provide several rules on the same topic: rules which perhaps should be implemented together within a task. Table 13 below shows an example of this.

if **if**	**If** you put a borehole down close to the springhead, you'll drawdown the water table in the vicinity of the springhead and cause it to migrate down the river system so you might be in a position where very little water is taken in a catchment overall but **if** you are having to put down a borehole at that location you could have quite a marked and very serious and unacceptable environmental effect in terms of drying up springs and the head of the river.	**If** a borehole is put down close to the springhead **then** the water table is migrated down the river system. **If** a borehole is put down at the springhead **then** there might be very serious and unacceptable environmental effects **because** springs and the head of the river may dry up.

Table 13: Example of multiple rules from a single sentence

The implementation of these 'paper' rules is described in Section 4.3.3 below.

3.7.4 Extracting relations from text

As with rule elicitation above, a 'glossary' of lexical semantic cues has been organised to facilitate the identification of relations between terms, and therefore objects. Once again these have been extracted from Quirk et al (1985) and from various thesauri of English. A sample of the cues used to indicate hyponym relations (the classical 'isa' relation) are shown in Table 14 below.

category	class	classification	collective * for
form of	forms of	group	include
including	is a	kind of	sort of
subcategory	subclass	subdivision	suborder
subordinal	subordinate	subtype	superclass
superordinate	taxonomy	type of	variety

Table 14: Examples of lexical semantic cues for hyponym relations

Examples of sentences found from a search on three of these cues, '*is a*', '*include*' and '*kinds of*' are given in Figure 7 below.

1_22225 A reservoir **is a** "raised reservoir" if it is designed to hold, or capable of holding, water above the natural level of any part of the land adjoining the reservoir, and a raised reservoir **is a** "large" raised reservoir if it is designed to hold more than 25,000 cubic metres of water above the level of the adjoining land.
1_19879 A public navigable watercourse **is a** public highway, and the owners of land on the banks are entitled to gain access to the watercourse from any point on their own land.
1_20088 A public ferry is a public highway of a special kind, consisting of an exclusive right to carry passengers across a river or arm of the sea from one place to another, or to connect a line of road leading from one place to another.
1_41262 1791 "Ditch" **includes** a culverted and a piped ditch but does not **include** a watercourse vested in or under the control of a drainage body.
1_17830 Hence, the meaning of "river" **includes** all natural streams, however small, which have a definite and permanent course, and excludes all bodies of water, however large, which are of a temporary character, that is, which are dependent on the will or convenience of individuals for their volume or duration.
1_17841 221(1) of the Water Resources Act 1991 states that "watercourse" **includes** all rivers, streams, ditches, drains, cuts, culverts, dykes, sluices, sewers and passages through which water flows except mains and other pipes which belong to the National Rivers Authority or a water undertaker or are used by a water undertaker or any other person for the purposes only of providing a supply of water to any premises.
1_17795 The expression "watercourse" is used to refer to a range of different **kinds of** moving waters, encompassing estuaries, rivers, streams and their tributaries above and below ground, which are commonly but loosely distinguished by characteristics of length, breadth and depth.

Figure 7: Example output from search for hyponym cues 'is a', 'include' and 'kinds of'

From this text a knowledge engineer could quickly produce a taxonomy of terms, or objects, as illustrated by Figure 8 below.

watercourse	river	stream
	ditch	culvert piped ditch
	drain cut dyke sluice sewer estuary tributary	

Figure 8: 'is a' hierarchy gleaned from the use of some hyponymic cues in Table 14

The implementation of object hierarchies in ELSIE is described in Section 4.3.1 below.

3.8 <u>Summary</u>

We have outlined the need to exploit the textual structure of specialist texts as a means of expediting the knowledge acquisition process. We have discussed at length how the role specialist terminology plays in scientific writing and the idiosyncrasies of scientific writing itself, can be easily exploited to identify domain objects, to find the salient attributes of the domain and to extract rules and problem solving tasks.

This methodology was operationalised through the use of a text and terminology management system, System Quirk. An important feature of our work is its *real worldliness* in that we were motivated to develop a methodology in the context of building an expert system that will eventually be used as an application program: ELSIE.

We would like to reiterate that we believe an understanding of how knowledge is encoded (and decoded) in text will perhaps overcome, in part, the knowledge acquisition bottleneck.

4. SYSTEMS DEVELOPMENT

4.1 The Components of ELSIE

ELSIE has several components, most of which are interactive with each other, with external data sources and with the user. WALDES (the Water Abstraction Licence Determination Expert System) is the central component of ELSIE, providing the user with guidance on whether or not a licence should be issued. WALDES covers the whole process of licence application, from initial inquiry to the NRA, through the statutory application procedure and on to the determination itself. Throughout the three month period in which a licence application must by law be processed, WALDES is continuously requesting data, gathering data, making decisions on that data and producing results with explanations to the user.

To assist WALDES throughout its life cycle, we have developed APMan (the Application Progress Manager). This system provides the essential link between the user and WALDES. It enables the input of requested data through a graphical interface mimicking the standard forms of the NRA, carrying out rigorous constraint checks. APMan also provides the means for passing back results when available, and keeping a track of when any new developments have occurred: it acts as a diary of events for the three month long process, letting the user know exactly what has or has not been done at any one time.

In order for the system to be used on many applications over such a long period, ELSIE interfaces to a relational database to store and retrieve the application specific data. The database interface is also used for retrieval of more general data within the domain, as and when WALDES requires it.

The licence application process necessitates a large amount of written communication between many people. ARGen (Automatic Report Generator) provides generic templates for many of these communications and incorporates application specific data into these templates when WALDES determines such a communication is required. ARGen also produces a technical report of all findings on which the technical officer will base his final report at the end of the determination period.

The process of licence application determination is strictly governed by the 1991 Water Resources Act of the British Parliament. This acts states under which conditions a licence may or may not be granted. ELSIE provides this act, and an interpretation of it, in a hypertext form through the Legislation Browser, enabling the user to search on common domain terms and swiftly consult referenced sections and definitions.

Figure 9 illustrates the components of ELSIE.

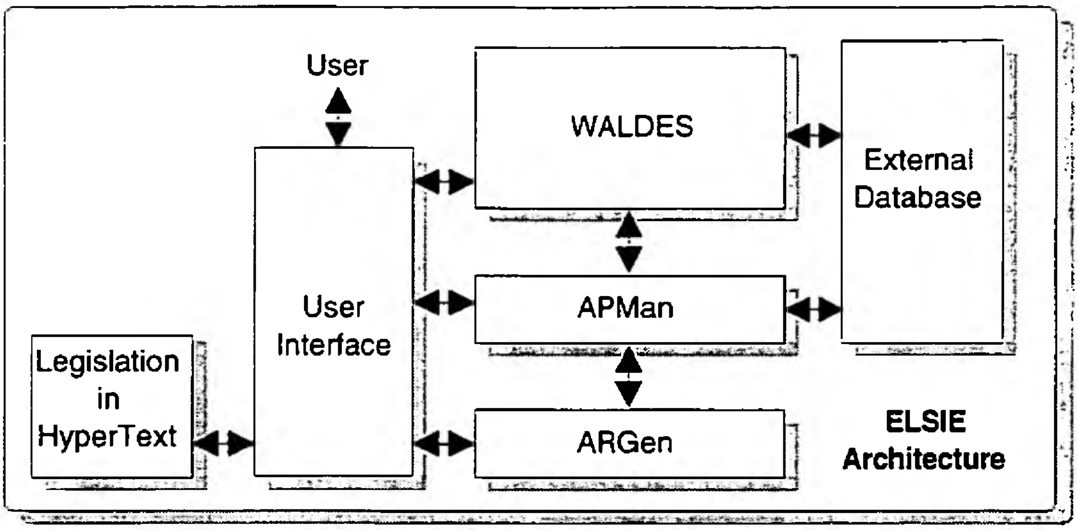


Figure 9: Components of ELSIE

Section 4.2 describes how ELSIE was designed around the tasks which a licensing officer undertakes throughout the determination process. Section 4.3 discusses how the paradigms of object-orientation, task-based reasoning and rule-based programming were used in the implementation of ELSIE, as well as a discussion on how the user and database interfaces were created. The implementation of the Legislation Browser is described later in Section 6.4.

4.2 Designing ELSIE

Two main design issues needed to be addressed very early in the ELSIE project: what the system should look like to the user and how the expert system knowledge should be structured. The issue of what the system should provide was addressed in a pre-project meeting of the project steering group and the developers in order to specify the objectives and terms of reference for the project, described in Section 2.

4.2.1 Designing the Appearance of the System

The design and subsequent development of a system's user interface is regarded by many as the most resource consuming task faced by system developers, taking up over 50% of their time and effort. The development team noticed very early in the project, however, that much of the data required by a licensing officer during the determination process is supplied in forms, which are now of a standard design throughout the NRA.

It was therefore put to the project steering group that the user interface of the system be based as closely as possible on these forms. This would serve two main purposes. Firstly, providing a familiar environment for the ELSIE user would help overcome the inevitable problems caused by the transition from a paper-based to a computer-based process. Secondly, with only main component windows and output screens to be designed from scratch, more resources could be put into knowledge acquisition and knowledge base development, resources which were already very limited by the length of the project.

Subsequently, Parts A and B of the national application form for the abstraction and impoundment of water, along with forms WR36 and WR38 from the new national test pumping manual were used as the basis for almost all of ELSIE's input screens. Part C was not included because the steering group decided early in the project that impoundment licences would not be covered by the system. Where data was required and no national form was available, the development team used regional forms supplied by members of the project steering group or based new designs as closely as possible on the existing formats.

Output screens were designed for fast and easy retrieval of data, providing results in easy to read forms, grouping the results from related tasks on the same output screen. Menus were provided to simplify the process of finding required information and these were made available from the main component windows.

The project steering group were requested to comment on the user interface throughout the design and implementation process and changes were made accordingly within the limitations of the ProKAPPA™ system in which the user interface was developed.

4.2.2 Designing the Expert System's Knowledge Base

In the W-RAISA project it was decided that the integration of a wide range of tasks within the same system, which was a likely requirement, required a structure that enabled the system and the human user to focus on a particular task under consideration. Each task represented a sub-problem which could be solved independently but which fitted into a set sequence of execution to provide a standard "guided consultation" through the determination process.

It was clear after preliminary analysis of the ELSIE interview transcripts that structuring of the ELSIE knowledge base was essential not only to aid the user but also to control the inference strategies which would be employed by the expert system. Over 200 candidate paper rules, as described in Section 3.7.3, had already been elicited, which if processed by a backward chaining inference engine, as in W-RAISA, would create a decision tree so large it could be impossible to control or debug. The development team again decided to take a task-based approach to the structuring and the knowledge engineer drew up a preliminary task hierarchy based on the contents of the interview on licensing procedure.

The task hierarchy is a tree structure which governs the order in which the licensing officer carries out the tasks necessary to determine a licence application. In order to process an application from start to finish, the tree must be traversed following a depth first search strategy: each task can only be completed on the completion of all of its subtasks and the same principle applying to each subtask.

The knowledge engineer created an initial task tree of almost 100 tasks into which the candidate rules needed to be positioned and invited the project steering group to a brainstorming session to agree on the tree structure and place the rules in the most suitable tasks. The group, with the knowledge engineer acting as a facilitator, spent over eight hours debating what the tasks should be and the structure which they should form.

The result of this meeting was a task hierarchy of over 200 tasks, forming a tree six levels deep. The complete structure is provided in Appendix A. This is a comprehensive breakdown of the tasks which a licensing officer must execute during the three months allowed for the determination of a licence and is exactly the structure of the knowledge base implemented in ELSIE.

There are six main tasks: pre-application, the application process, pre-determination, technical assessment, determination and post-determination , as shown in Figure 10.

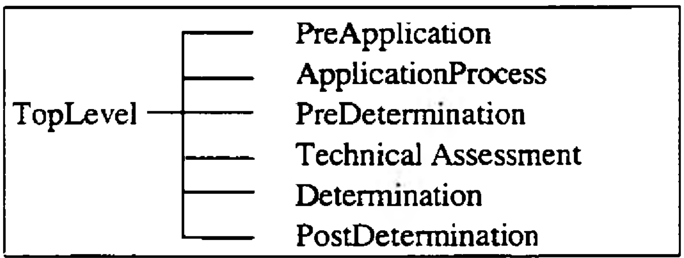


Figure 10: Top level of the ELSIE task structure

As an example of the tree structure, Figure 11 shows the structure of the source of supply task, a subtask of technical assessment. Numbers of the paper rules elicited from the interviews, in the format V(video number)R(rule number), are provided where applicable. Appendix B provides a few of the original paper rules used in the task base.

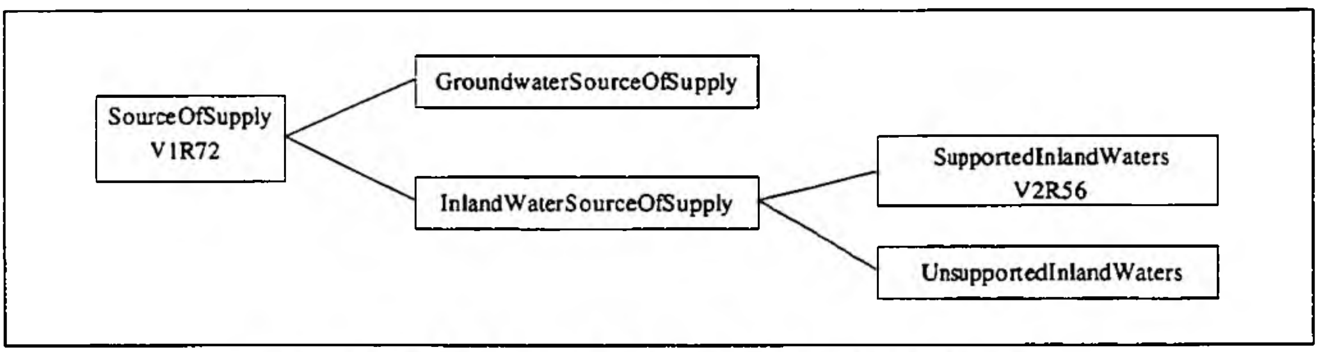


Figure 11: The Source of Supply task and its subtasks

The implementation of the task hierarchy is described in Section 4.3.2 below and the implementation of the paper rules is discussed in Section 4.3.3 below.

4.3 Developing ELSIE

WALDES, the central component of ELSIE, is an expert system, and as such has four main underlying components: a knowledge base, an inference engine, a user interface and an interface to external data sources. APMan requires all these components except an inference engine. The choice of representation schema to use for the knowledge base is, in any expert system project, a critical design decision.

ELSIE required a hybrid system encompassing object-oriented programming techniques for the large scale repository of domain knowledge and production rules which could be structured into a large task base to be reasoned over. It also required tools for the development of a graphical user interface and components to enable communication with relational databases.

ProKAPPA™, from IntelliCorp Inc., is a software development environment encompassing most of these features, requiring only the in-house development of meta-level knowledge representation, such as the combination of the object and rule paradigms for the implementation of large scale task hierarchies.

Figure 12 illustrate the main components of an expert system as used in the ELSIE development.

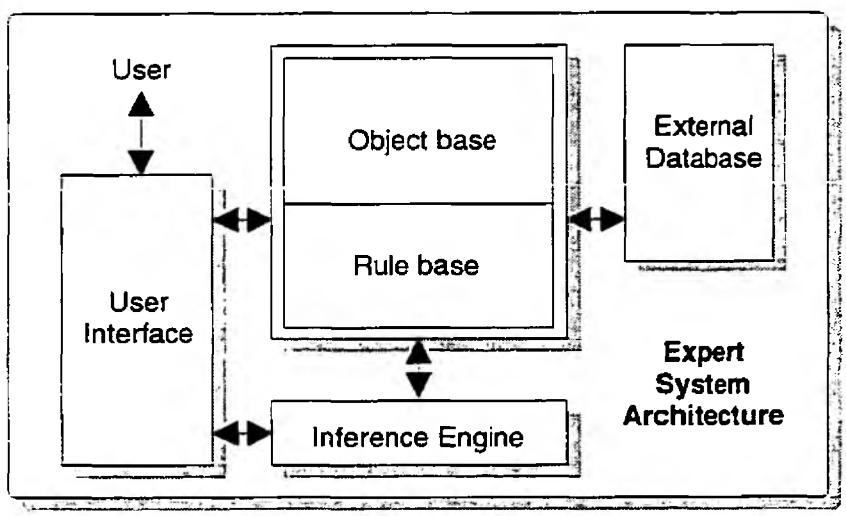


Figure 12: Main components of an expert system

The ELSIE knowledge base can be broken down into an object base for storing data and a rule base for reasoning over this data: the inference engine provides a mechanism to control the reasoning process.

The user interface provides the user with facilities to add and modify input data and retrieve system generated data and the interface to external data sources provides a means of retrieving data from, and depositing data to, an external relational database.

4.3.1 Object-Oriented Knowledge Representation

All computer systems work on data. In conventional (algorithmic) tasks, data items do not have complex relationships with each other, or if such a relationship exists, it is hierarchical and unidirectional (as in data-base management systems). Where data items are linked in complex patterns, as is common in expert systems, this requires representations which allow for such relationships. Various structures such as frames and semantic nets have been proposed to deal with this type of situation; however, the meaning attached to such structures varies a great deal from one source to another. We use the term object to refer to a way of aggregating several simple logic-based data statements into one structure which is strongly associated with an object of the subject domain.

The ELSIE object base consists of a set of classes, prototype objects describing the properties or characteristics common to the class. Examples of classes would include *CatchmentDetails* and *GroundwaterSourcePoints*.

The properties of a class are described by its slots: pigeonholes for data within the structured object base. Examples of slots for the *CatchmentDetails* class are *Name*, *Area* and *ExistingWaterBalance*. The data stored in these slots in a specific instance of a catchment would be its name, area and the name of the object storing data relating to its existing water balance. The slots in turn have facets which describe the attributes of the slot, such as which values it is allowed to take etc.

Some slots do not contain data but methods: functions performing actions specific to a particular class. Slots may have monitors attached to them, hooks which call methods when a particular trigger is fired, such as when the value of a slot is required or changed.

A detailed description of object-orientated knowledge representation can be found in R&D Project Record 406/3/S.

4.3.2 Reasoning with Task Hierarchies

The tasks defined during the design of the ELSIE system, discussed in Section 4.2 above, have been implemented as objects within the ProKAPPA™ system. A class was created called *Tasks* which contained slots including *Name*, *Description*, *Complete*, *Start*, *Successor*, *EntryAction*, *Goal*, *RuleSet*, *ReasoningStrategy* and *OKToBrowse*, among many others. Details of the information stored in these slots can be found in R&D Project Record 406/3/S.

We have implemented a function, stored as a method called *Execute!* in the task class, which oversees the execution of tasks based on the information stored in these slots. This function handles the starting of subtasks and successors in the task hierarchy, informs the user at all times whereabouts in the task hierarchy he/she is, and can also start and stop task hierarchy traversal at any user designated task. Details of how the *Execute!* method works can be found in R&D Project Record 406/3/S.

This task based reasoning strategy allows enormous control and flexibility to the user wishing to access the knowledge base. Any of the tasks specified in the hierarchy can be executed, however big or small, at the top or bottom of the tree structure. This enables novices to use the system for specific problems which they have not seen before and experts to use the system to maybe remind them of the requirements for a rare task or verify what they have done at any stage of the determination process.

4.3.3 Production Rules

Rules were introduced in Section 3.7.3 where we discussed how they may be elicited from the interview transcripts, and Section 4.2 described how they were placed into a task hierarchy to produce the paper knowledge base of ELSIE. Here we describe how these paper rules are implemented in ProKAPPA™.

Our task implementation, described in Section 4.3.2 above, requires that all rules to be used in a single task be part of the same ruleset. The ruleset is a useful feature of the ProKAPPA™ rule system as it enables the specified set of rules to be linked separately into a decision tree by the inference mechanism.

Details of the rule-based reasoning mechanism and the creation and traversal of decision trees can be found in R&D Project Record 406/3/S, but as an example consider the ruleset for the *SourceOfSupply* task. The implemented rules for this task are shown in Appendix C: note that the few rules provided in the paper knowledge base are found in subtasks but during implementation many small tasks like these were subsumed into the parent task, in this case *SourceOfSupply*.

Figure 13 shows the decision tree created for this single task.

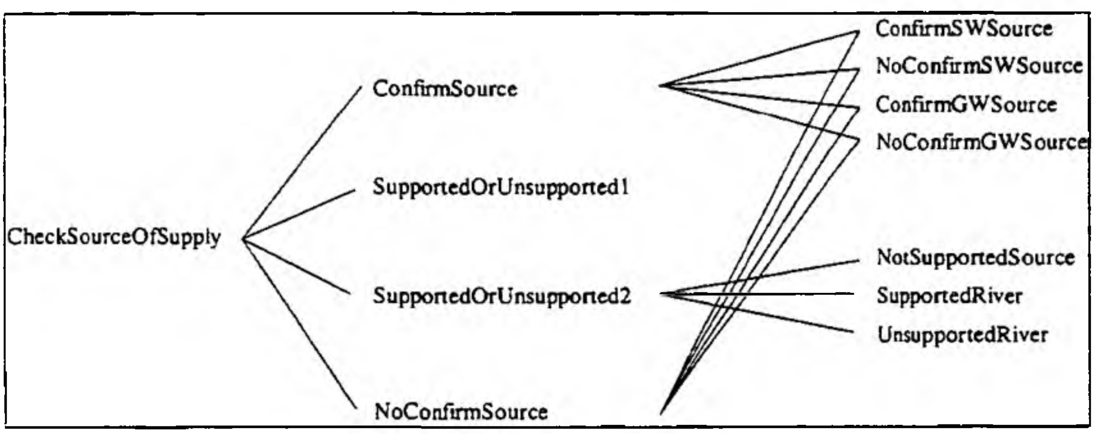


Figure 13: Decision tree generated for the *SourceOfSupply* ruleset

Figure 13 shows that when the task *SourceOfSupply* is executed up to twelve rules may be fired. The first is the head rule of the set, *CheckSourceOfSupply*. This is because the consequent of this rule is that all sources have been checked which is the goal of the task. The backward chaining mechanism then searches for rules with consequents which match the antecedant of this rule and finds the four rules *ConfirmSource*, *SupportedOrUnsupported1*, *SupportedOrUnsupported2* and *NoConfirmSource*, listed with priorities from highest to lowest.

Not all of these four rules need necessarily fire. For example, if *ConfirmSource* and *SupportedOrUnsupported1* both fire and complete successfully then the *CheckSourceOfSupply* rule's antecedant is satisfied and therefore its consequent is satisfied making the task complete. If, however, one of these rules is not satisfied, the other rules will be fired in order to find one that is. The rules to the right of these four rules are linked to these rules following the same principle, and again may fire or not depending on the outcome of the rules with the highest priority.

This shows how the complex chaining algorithms of the built-in inference mechanisms are kept to a manageable size by the implementation of the higher level task structure and the use of rulesets within these tasks.

4.3.4 Interfacing with the User

The ELSIE interface was developed entirely within the Graphical User Interface (GUI) builder of ProKAPPA™. This enables fast and straightforward creation of GUI components, such as dialog boxes, by allowing the knowledge engineer to draw the form on a palette. The 'drawing' is then converted into windowing component objects which are implemented in exactly the same way as domain objects described in Section 4.3.1 above, but exhibiting properties such as *Height*, *Width*, *BackgroundColour* and *ForegroundColour* etc.

Communication with the forms is via methods in the window objects, which display text on the screen. Functions were written to link the forms with specific domain objects to enable two-way passing of data from the domain object to the form and vice-versa. This enabled us to use the monitoring of slots, as described in Section 4.3.1, to verify data being input by the user.

For example, when the user adds a new value or changes a value of a slot through a form, a 'When Changed' monitor on that slot may check that the new value is allowed and if not a message can be passed to the user. This is essential for the consistency checking capabilities of APMan, where 'When Changed' monitors can check that entered values are consistent with previously input data and correct it if necessary whilst the form is still on the screen.

Further consistency is enforced by the use of pull-down menus of allowable values for as many input boxes as is opssible. The system is written to handle a fixed number of possible values for each piece of information and the use of these menus ensures that a known value is used.

The GUI developed can be seen in the discussion of the ELSIE system in Section 5.

4.3.5 Interfacing with External Databases

ProKAPPA™ provides a subsystem to handle the transfer of data from an Oracle® database to the object base and vice versa. This subsystem, consisting of several C functions, was used to create all interfaces from ELSIE to external databases.

5. THE EXPERT LICENSING SYSTEM AND INFORMATION ENVIRONMENT (ELSIE)

This section describes the workstation components of ELSIE: the Legislation Browser, although still part of ELSIE, is available only on PC and is described separately in Section 6. We present here an overview of ELSIE's operation, a more detailed description of which can be found in the ELSIE User Manual 406/2/S.

5.1 General Information

The user interface of ELSIE is based almost entirely on the various forms viewed regularly by licensing officers, such as the National Water Abstraction and Impoundment Application Package used by the applicants when they apply for a licence and the forms used in the Groundwater Investigation Consents manual.

The user can use one of the two principal ELSIE components: the Application Progress Manager (APMan, see Section 5.3) and the Water Abstraction Licence Determination Expert System (WALDES, see Section 5.4). APMan is intended to act as an electronic version of a busy executives diary in that the diary contains reminders for the executives relating to pending and forthcoming tasks. APMan not only contains the reminders but has access to the details of each application. WALDES provides access to various tasks ELSIE can execute, as shown in Appendix A.

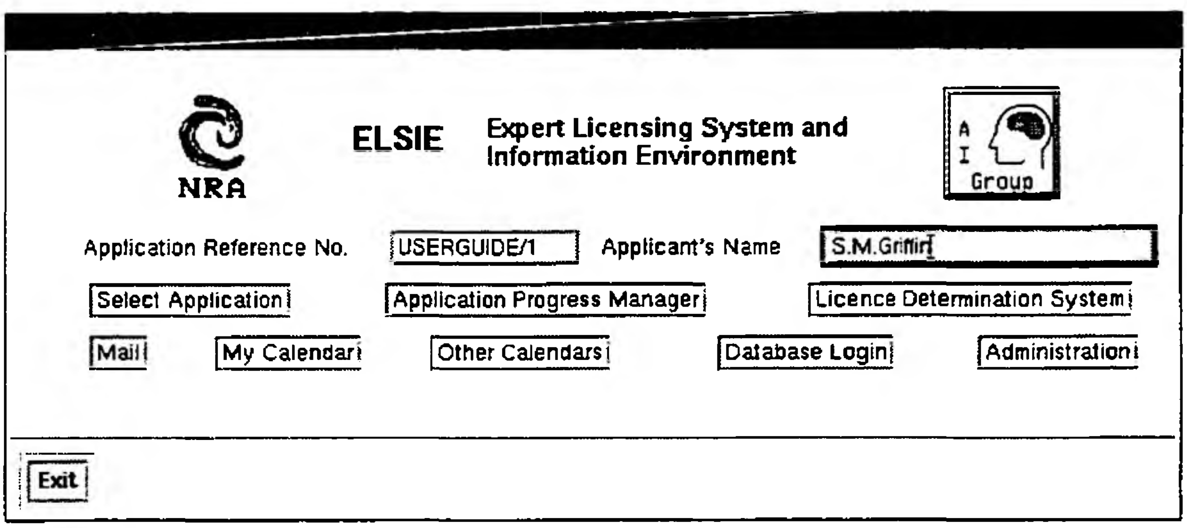


Figure 14: The ELSIE Main Window

From the ELSIE main window the user may select an application to load from the database or to clear the ELSIE system for a new application by pressing the "Select Application" button (see Section 5.2) .

The user may start up APMan (Section 5.3) and/or WALDES (Section 5.4) by selecting the "Application Progress Manager" and "Licence Determination System" buttons respectively. As we shall see in Section 5.4, WALDES can be executed as an expert system or set to 'browse mode': this mode enables a novice user to traverse all or any part of the task hierarchy, not executing any tasks but simply browsing the information attached to each task in paper knowledge base form. This is ideal for novices who are seeking an overview of the licensing officer's role in the licence determination process.

This main window also gives the user access to a mailtool, his/her calendar and calendars of his/her colleagues. These tools are delivered with the operating system (Open Windows) on which ELSIE was developed.

The "Database Login" button accesses the ORACLE database where all previous data used in ELSIE consultations is stored. The database must be logged into before an application may be retrieved or saved to the database.

Finally, the system administrator may make changes to the system's user details using the "Administration" button.

5.2 Select Application

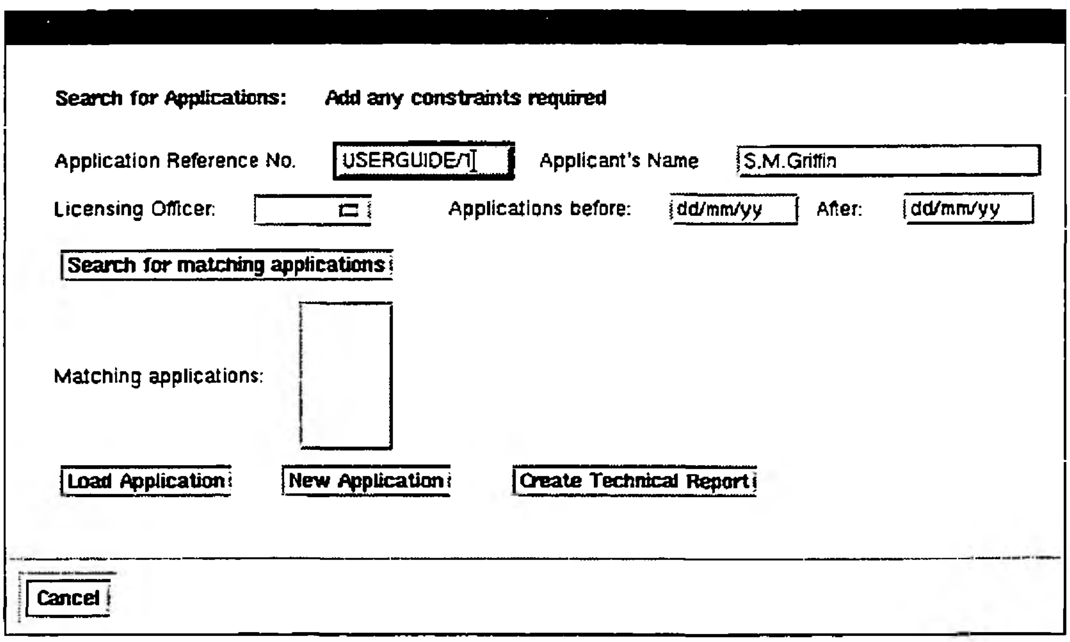


Figure 15: The 'Select application' dialog

This is a facility for searching through an Oracle database for applications which match criteria, or "constraints", specified by the user. For example, a user might wish to search for an application which was submitted by a Mr S.M. Griffin, reference no. USERGUIDE/1. This data is inserted in the relevant boxes and a search is made for the application which matches the criteria. The wildcard % may be used in any text box to denote "any characters", thus USERGUIDE/1 may be searched for with USER%.

Note that even with only very little information, the system usually finds the relevant application successfully. If there is more than one match for the criteria, which is often the case when wildcards are used, a list is presented to the user to select the appropriate application. When the application required is listed, the user simply highlights it in the list and presses the "Load application" button.

The user may wish to only print a technical report based on the data in a stored application rather than load the entire application into the system. In this case, the application is searched for and selected as above, but the user then presses the "Print technical report" button instead of "Load application".

The select application window is also used when a user wishes to start adding a new application to the system. In this case, no searching needs to be carried out, since the application will not be found. The user simply presses the "New application" button which will empty the system of its current application (which may be saved on entry to the Select Application subsystem) and, after asking the user for the new application number, sets up the system for the new application to be entered through APMan.

5.3 Application Progress Manager (APMan)

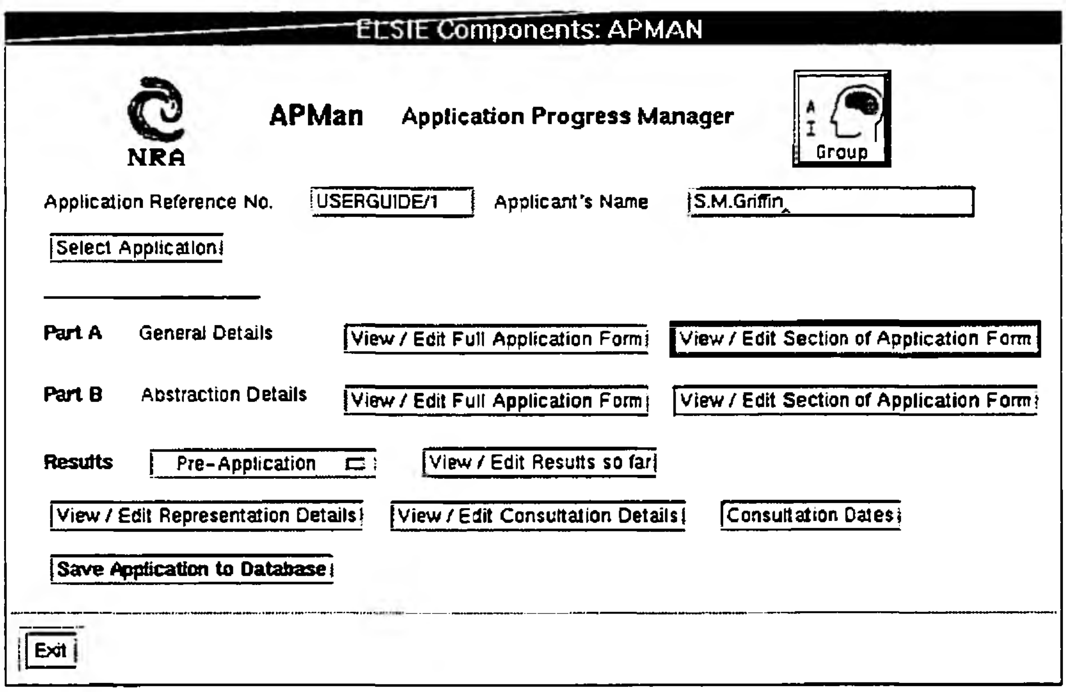


Figure 16: The APMan main window

APMan allows the user to view or edit application forms for licenses to abstract and/or impound water. The application form template is divided into Part A and Part B. Part A is concerned with general details of the application. Part B concerns information regarding the proposed abstraction. The user can choose to view/edit the full application form or to view/edit sections of the application form.

If the full application form is requested by the user, it appears on screen in the form of a consecutive series of windows. These windows request information from the user with regard to the application; some of the information must simply be typed in, other types of information for which only a limited number of values are relevant (such as "yes" or "no") can be entered using the "toggle" facility.

If the user wishes to fill in sections of the application form at a time, "View/Edit Section of Application Form" should be selected. The "Chooser" menu (shown below) then appears on the screen.

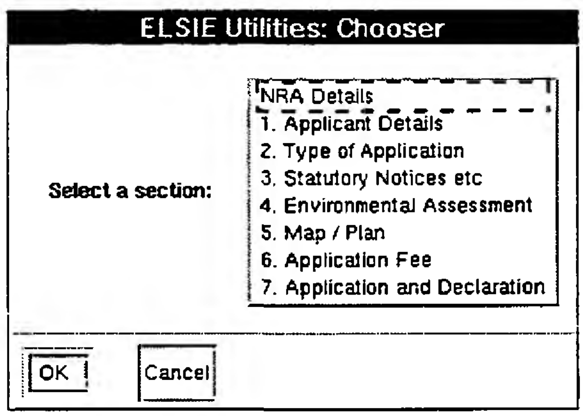


Figure 17: View/Edit Section of Application Form (Part A)

The user selects the relevant section by simply highlighting the section and pressing OK. Once a section has been selected, the "Chooser" menu will disappear from the screen. To retrieve the menu in order to make another selection, the user must return to the APMan window and reselect "View/Edit Section of Application Form".

Some sections of Parts A and B is shown below. The user also has the option from the APMan window to view/edit results of the current WALDES consultation, representation and consultation details, including the dates consultation details were sent and comments were received, by pressing the relevant buttons at the bottom of the window. Results windows are described in Section 5.5 below and representation and consultation details are described in under WALDES (Section 5.4) where they are most frequently used.

A note on "Commit Changes" and "Done"

The user will notice that on the following and many other windows, there are two buttons: "Commit Changes" and "Done". Once the relevant data has been entered in a window, the user should press the "Commit Changes" button to save the data. Having saved the data (and presuming that all the relevant information has been entered in the window), the user should press the "Done" button in order to close the window, that is remove it from the screen. Note that if the user does not save the data before pressing the "Done" button, any new data or edits are lost. In this event, the user should reselect the relevant section from APMan and re-enter the data, remembering to save the data before closing the window.

Figure 18 shows the NRA details form of the application package and Figure 19 the applicant details form. These figures illustrate how closely the dialog boxes of the user interface match the original paper forms.

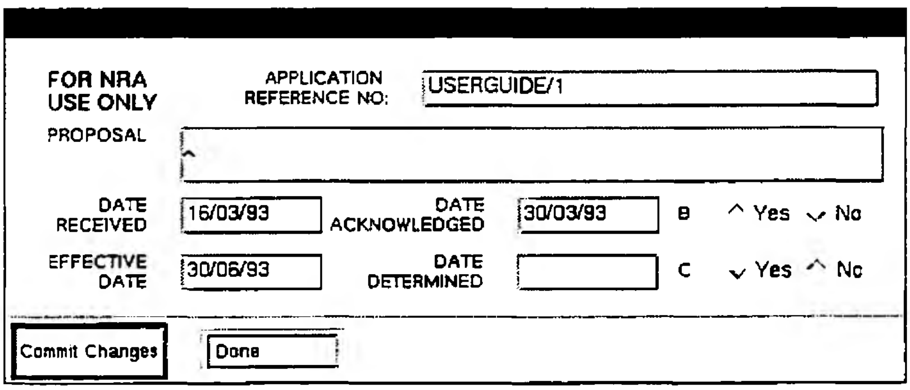


Figure 18: NRA Details input form in APMan

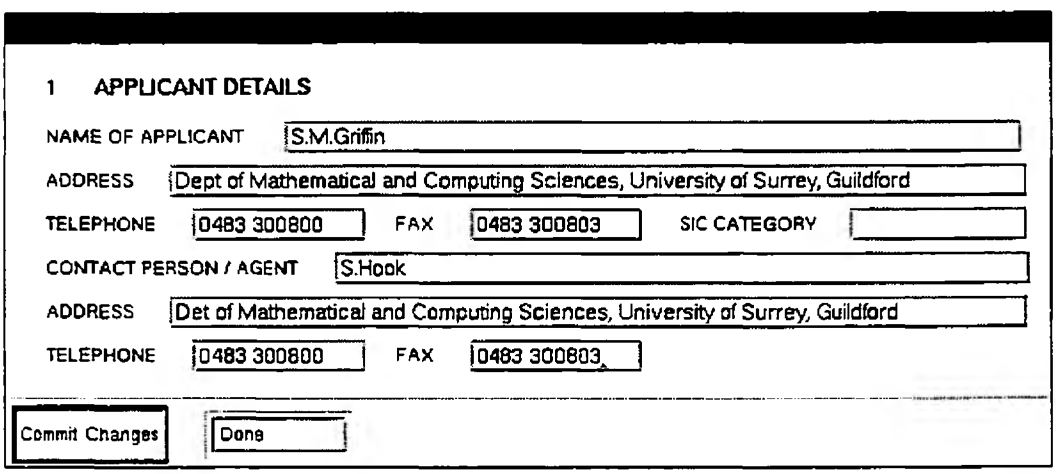


Figure 19: Applicant Details form in APMan

As stated earlier, Part B of the application form is for information specific to abstraction and may be selected from the APMan window. The user may view/edit Part B in full, or may view/edit individual sections of Part B. The appropriate section(s) may be accessed from the "Chooser" menu shown below.

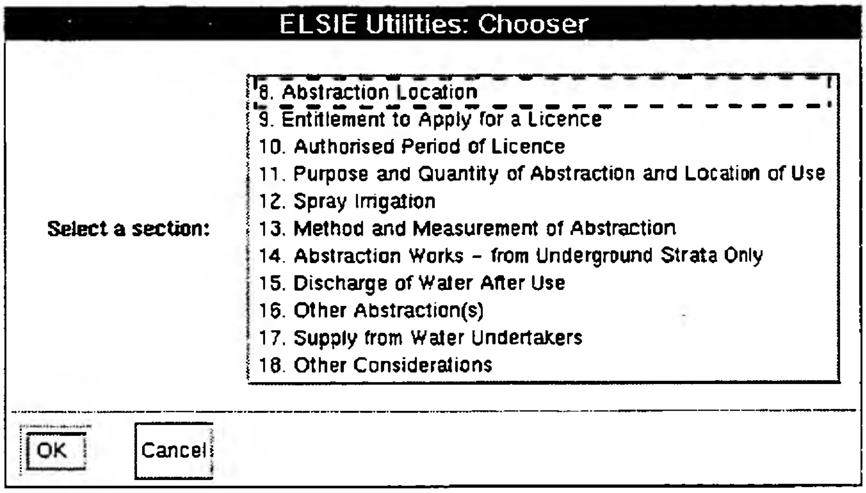


Figure 20: View/Edit Section of Application Form (Part B)

The following screens are examples of forms from part B of the application package. Note that these forms do not have to be completed in order since APMan provides ongoing consistency checks between the data being added at all times. For example, if the user were to not add an abstraction location by filling in Form 8 (Figure 21 below), but then attempted to add an abstraction purpose (Figure 24), APMan would insist on Form 8 being completed first because it makes no sense to have a purpose for an abstraction which has no source.

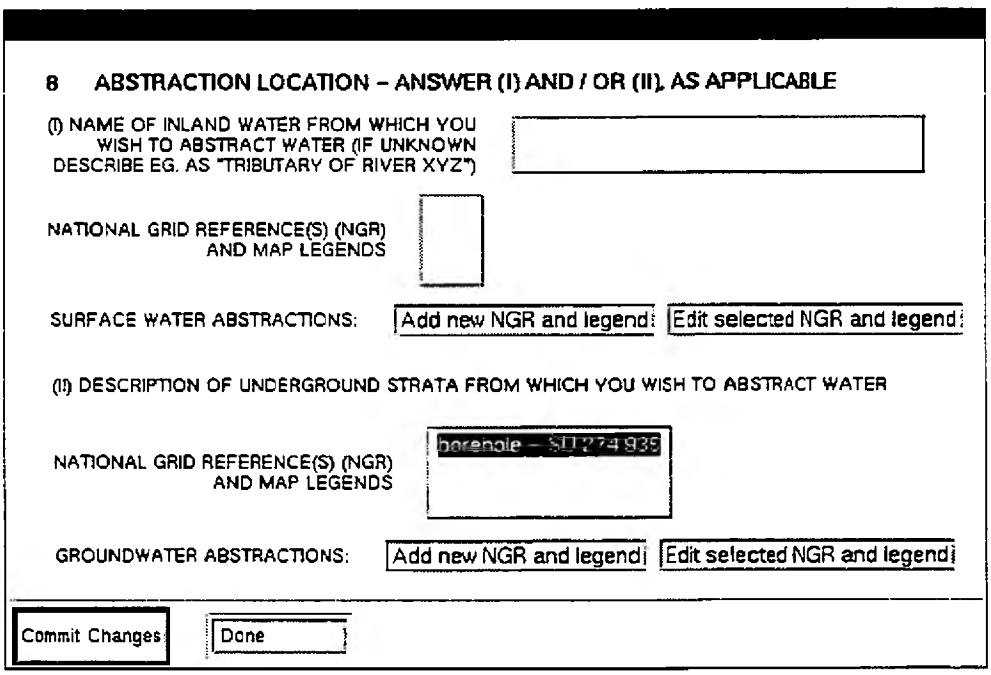


Figure 21: Abstraction Location

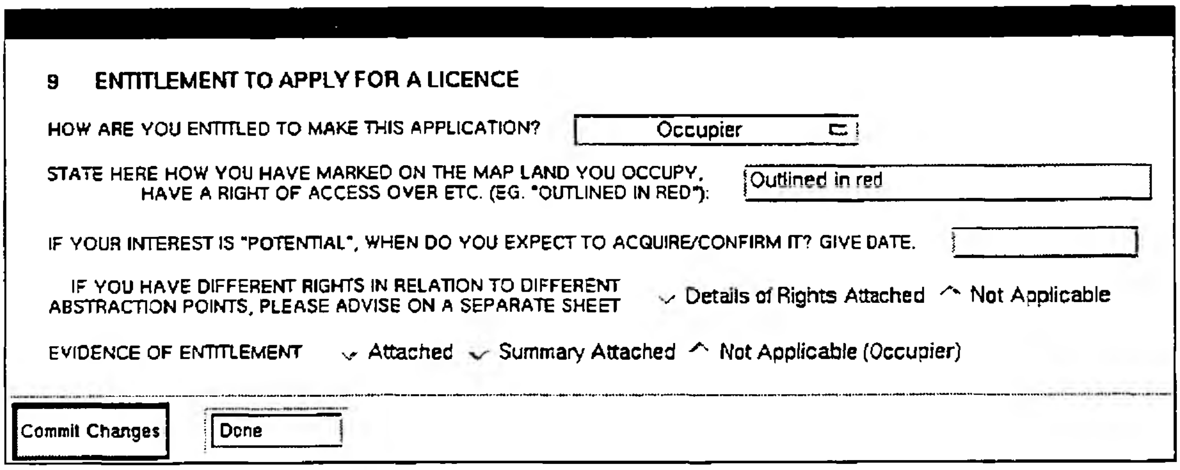


Figure 22: Entitlement to apply for a licence

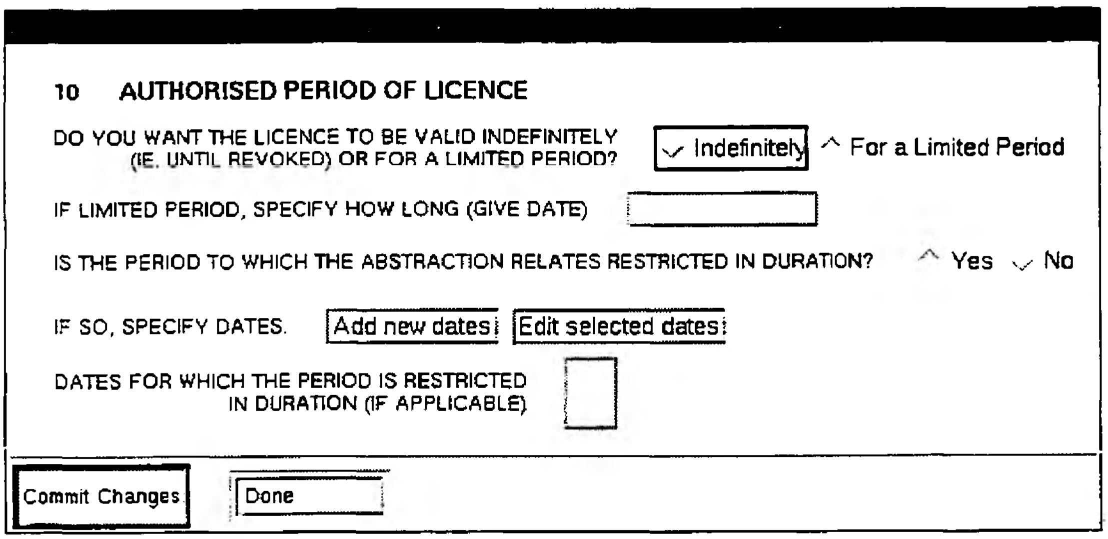


Figure 23: Authorised Period of Licence

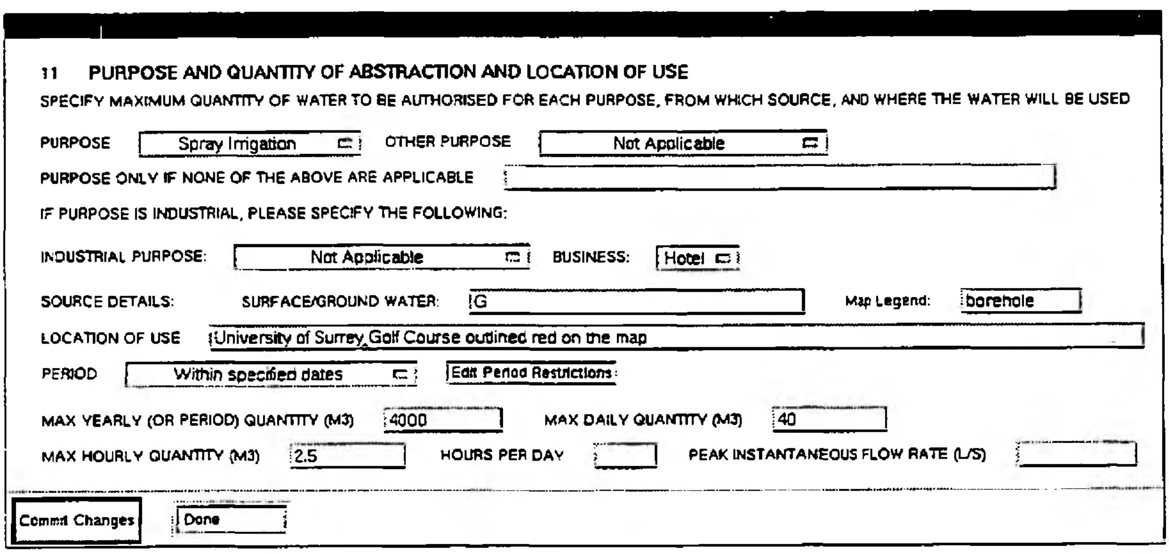


Figure 24: Purpose and Quantity of Abstraction and Location of Use

Note here that certain responses in a window cause the system to prompt for more information. For example, if the user selects the option "Within specified dates" for the period of the purpose, a new window appears on the screen requesting further details of these dates (shown below). Also, committing changes for a section may have a "knock-on" effect. For example, if "Spray irrigation" is selected for the purpose then on committing changes the Spray Irrigation section of Part B is presented to the user for further details.

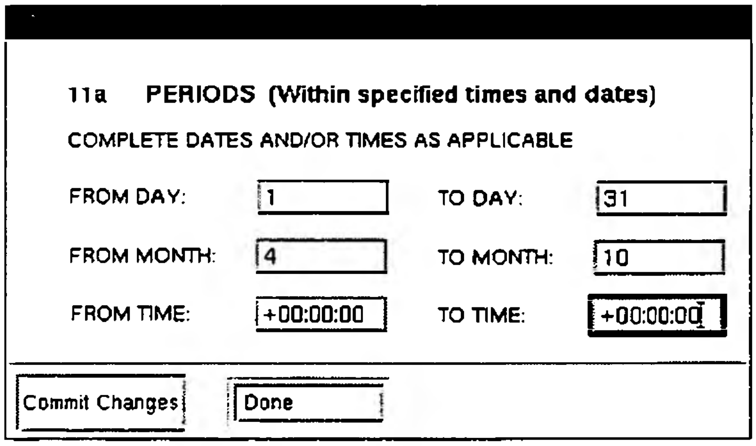


Figure 25: Period Details

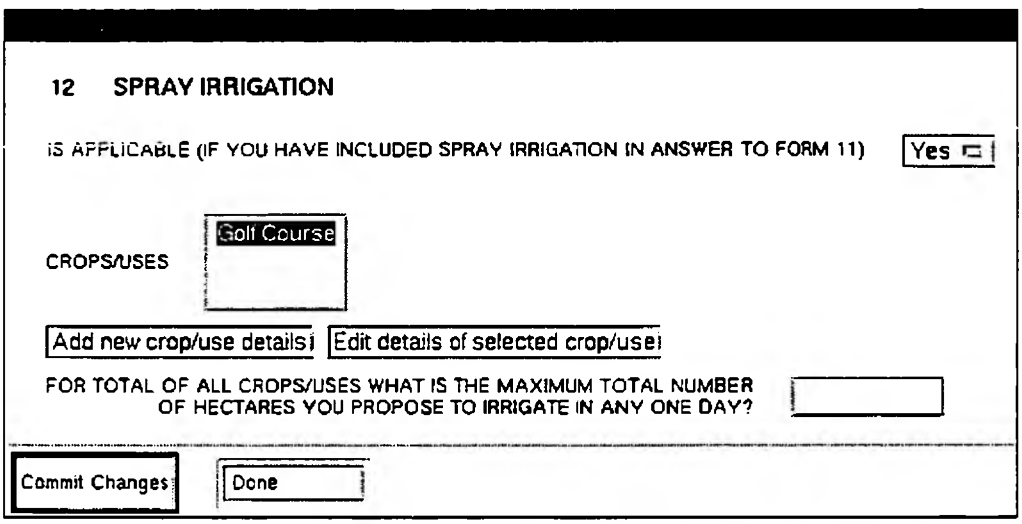


Figure 26: Spray Irrigation form in APMan

Another point to note is that many forms contain lists of all possible responses to a question on the form. For example, Section 11, the Purpose of the Abstraction, lists all purposes mentioned on the paper version of the form and a breakdown of industrial purposes to cover all those described in the NRA Charging scheme. Similarly, Section 12, Spray Irrigation Details, contains an extensive list of crops to be irrigated and other uses such as golf courses.

Part C of the form was not implemented in APMan because at an early stage of the project the Steering Group decided that ELSIE should not cover Impoundment Licences due to the time limitations on the project.

Some data needs to be added to ELSIE at installation in each region. This includes details of the catchments of the region, rivers within the region and any stretches of rivers which may be supported by the NRA. However, APMan does allow for catchments and rivers to be added if necessary and their details to be taken, although at present this information is application specific and is not saved to a central database.

5.4 WALDES

WALDES is the Water Abstraction Licence Determination Expert System: a production rule- and frame based system to aid the Licensing Officer in the many tasks he / she is required to perform during the three months in which a decision on whether to grant a licence or not must be made.

WALDES has over two hundred tasks arranged in a sequence to guide both expert and novice through a wide range of concerns, from checking that the application form is correct, through consultations and representations, technical assessment including groundwater and surface water investigation, and on to helping with the final choice of conditions for a licence.

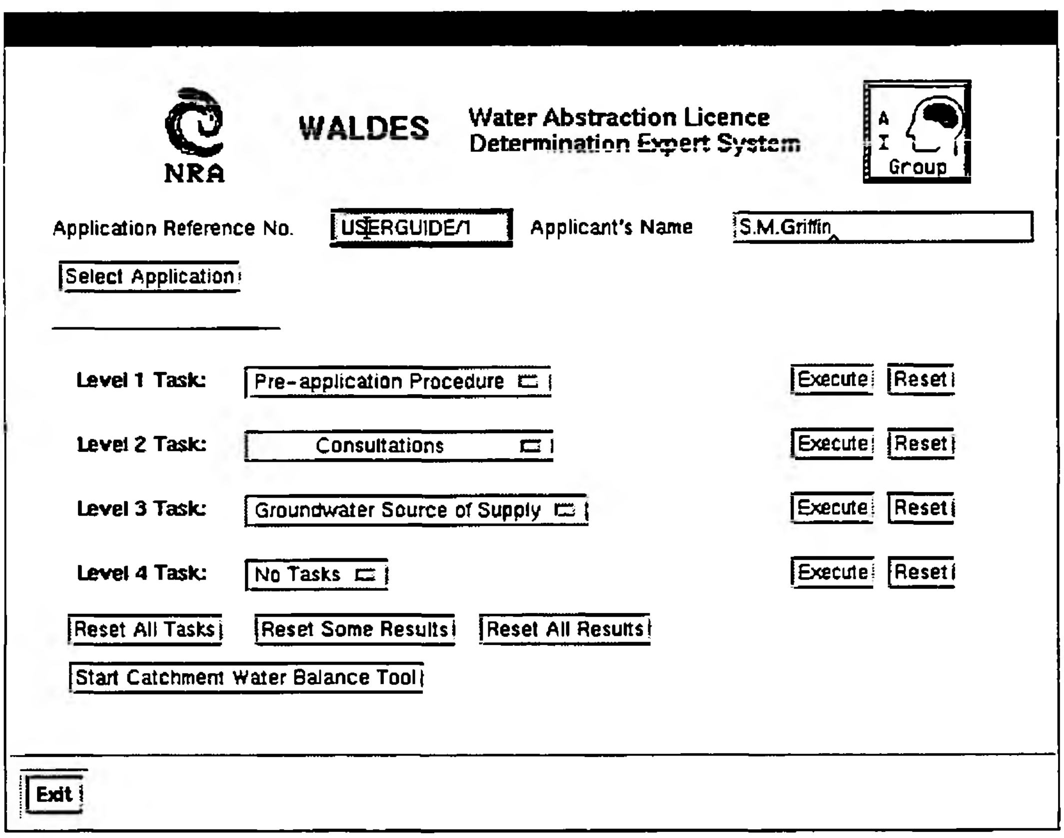


Figure 27: WALDES Main Window

The "Select Application" button takes the user to the same screen as the identically titled button on the ELSIE main window (see Section 5.2). To choose a task to execute, the user must first select the required top level (level 1) task from the top option menu. When a task has been selected here, all of its subtasks are placed in the option menu below it, and so on for up to four levels of tasks.

To execute a task, simply press the execute button alongside the task required. If the selected task has subtasks, these will be executed automatically by the system when needed. The user should be aware that executing a level 1 or 2 task may take a long time and not all tasks have a way of escaping part way through. Level 3 and 4 tasks are usually much shorter, and can usually be executed separately before their super tasks. The super tasks will then take less time to execute because most of the data they require has been collected already.

The "Browse Mode" feature enables a novice user to browse through any part of the system's task structure without executing a single task. When browse mode is on the user will be presented at each step with a "paper" form of the rules used in the implemented version. When browse mode is switched off again, implemented tasks will execute as normal. This feature is extremely useful for novice users: it provides the user with an overview of the role of the licensing officer and the tasks he/she must perform at all stages of the licence determination process.

The reset task buttons force the system to reconsider a task which may have been competed already: this may be useful if you find you had the wrong data the first time, or some new data has been made available. It should be noted, however, that just resetting the task may not force a re-execution, because the data required to complete the task may still be stored in the knowledge base. At present, the user then has only the option to reset the results for the top level task being executed. This can be achieved by pressing the "Reset some results" button and selecting the top level task you are executing, or the whole of the system's results may be reset using "Reset all results". It is envisaged that in the future only data relevant to the smallest tasks may be reset before re-execution.

5.4.1 Examples of WALDES screens necessary for every application

Most applications attract at least a few representations: this window is used to take details of representors from the user who will be receiving the representations through the post, and to keep him / her informed of the validity of the representation: a valid representation may become important during the determination process .

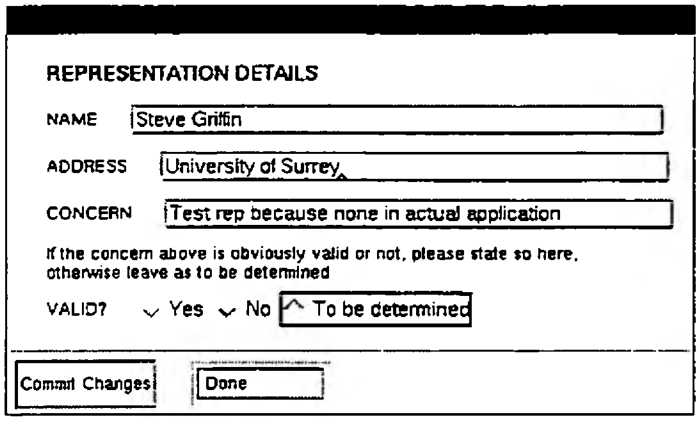


Figure 28: Representation Details

Some consultations are mandatory and some optional, depending on the application. This window informs the user which consultees have been sent details of the application and, if they have, when the details were sent. It also states whether or not the consultee has responded and, if so, when they did. If the user wishes to view more details on a consultee, or add a response just received, they must simply select the consultee from the appropriate list and press the corresponding button. The Consultation Details window below will then be displayed.

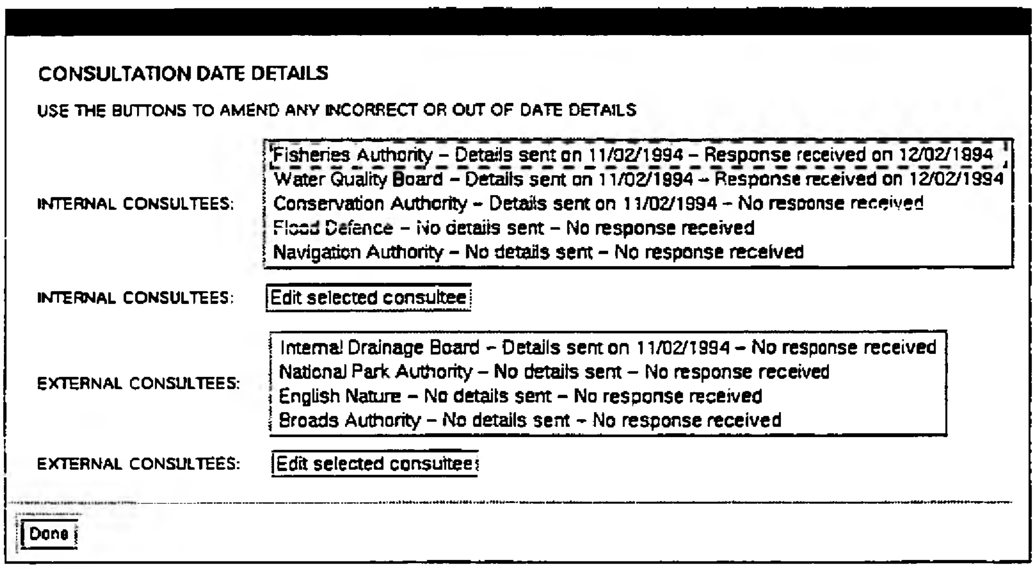


Figure 29: Consultation Dates

The load and save details to database buttons enables details of regular consultees to be stored and retrieved to a regional database if present. This again would be extremely useful for novices who do not know the contacts at consultee establishments.

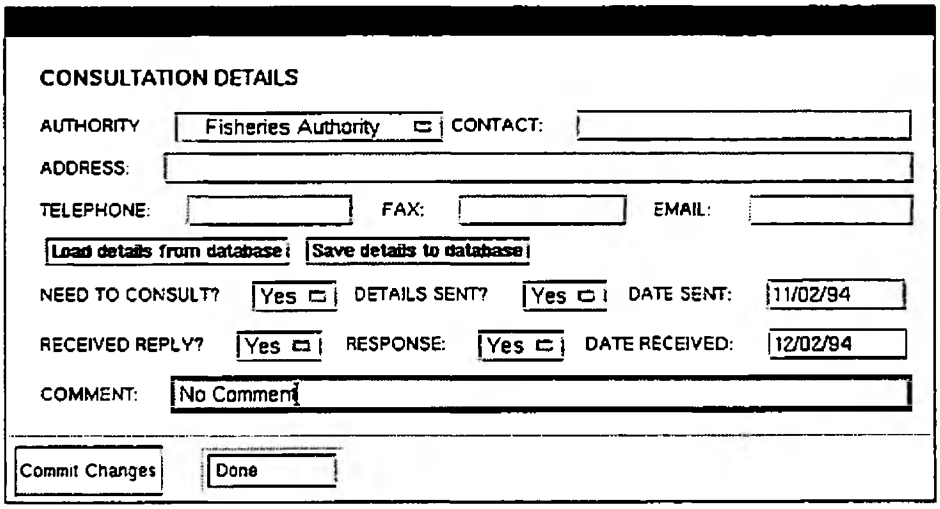


Figure 30: Consultation Details

5.4.2 The Catchment Water Balance Tool

The catchment water balance tool enables the user to edit catchment water balance values for catchments in the application, to create new, often hypothetical, catchments for testing and training purposes, and to view charts illustrating the components of the water balance and whether or not any water is available at present and / or after the application has been added.

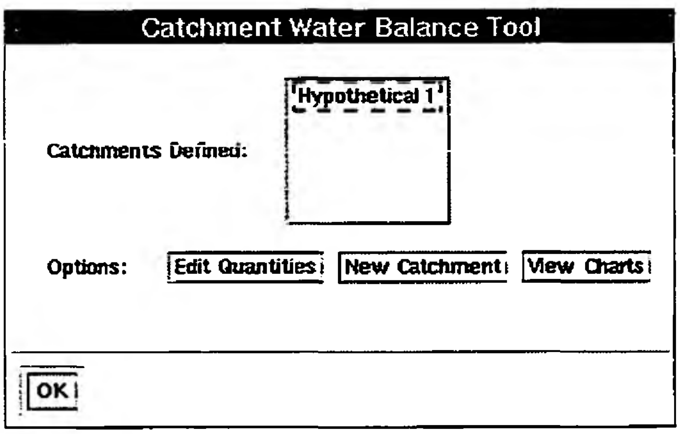


Figure 31: Catchment Water Balance Tool

Selecting the "Edit Quantities" button displays the following window for the user to input either daily, monthly, yearly or any other period quantities for the components of the water balance. It is important to stress, however, that whichever period is used it should be consistent throughout the form! The "New Catchment" button enables details of any number of (usually hypothetical) catchments to be added to the system. The "View Charts" button displays cachment water balance data graphically to the user, including a breakdown of water entering and leaving the catchment in pie charts and a fan chart showing how much water is available for new abstractions. The user can also view the catchment water balance taking into account the proposed abstraction to ensure there are still resources available if a licence is granted.

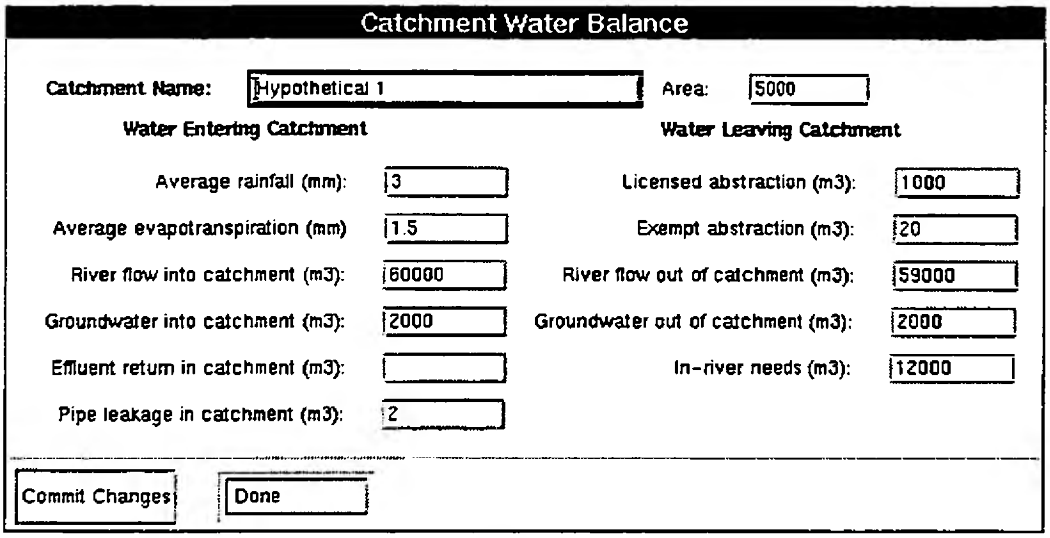


Figure 32: A Hypothetical Catchment Water Balance

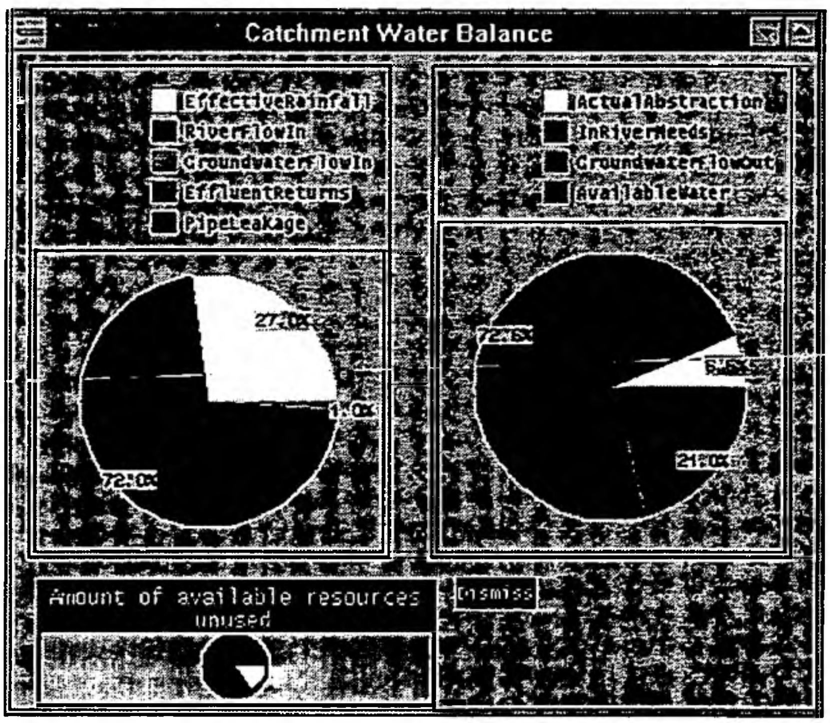


Figure 33: Graphical Display of a Catchment Water Balance

5.4.3 WALDES screens which may be necessary in an application

The following screens are exemplar of those which may be presented to the user at some stage of a consultation with WALDES. Which screens are seen in any one consultation depends on many factors: for example the minimum water level and prescribed flow input screens may not be requested for a groundwater abstraction. More screens of WALDES are described in the ELSIE User Manual 406/2/S.

It is foreseen that in the future the NRA will have a National Database which will store data on all catchments and rivers etc. ELSIE does not, as yet, look for this data in any database but asks the user for all such information. Two examples of this can be seen below.

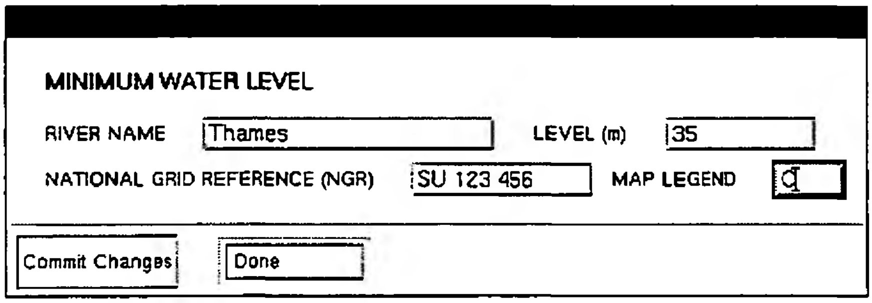


Figure 34: A Minimum Water Level

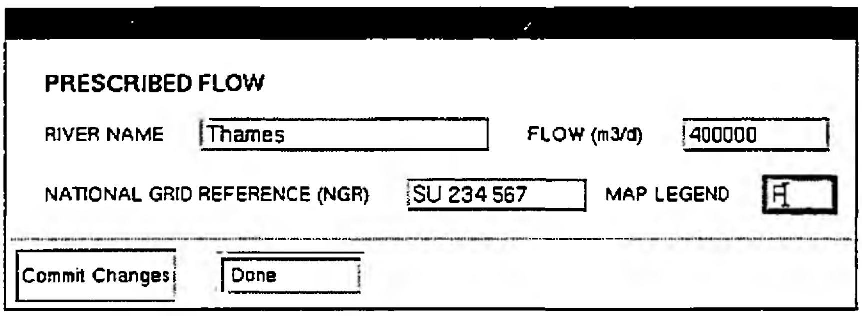


Figure 35: A Prescribed Flow

These screens ask for minimum water levels and prescribed flows at specific National Grid References along the river which is the source of the proposed abstraction. Both minimum water levels and prescribed flows are considered in the Surface Water Investigation task of WALDES, where the current measured levels and flows and the proposed abstraction volume are considered to estimate the future levels and flows which are then compared with the set minimum and prescribed flows for the river.

Another screen for gathering data which may one day be available via databases is the Survey Report Form WR36 from the Goundwater Investigation Consents manual. This screen collects details on all nearby interests to the source of the proposed abstraction, including nearby licensed abstractions (as shown in Figure 37), nearby exempt abstractions, nearby surface waters and nearby Sites of Special Scientific Interest (SSSIs).

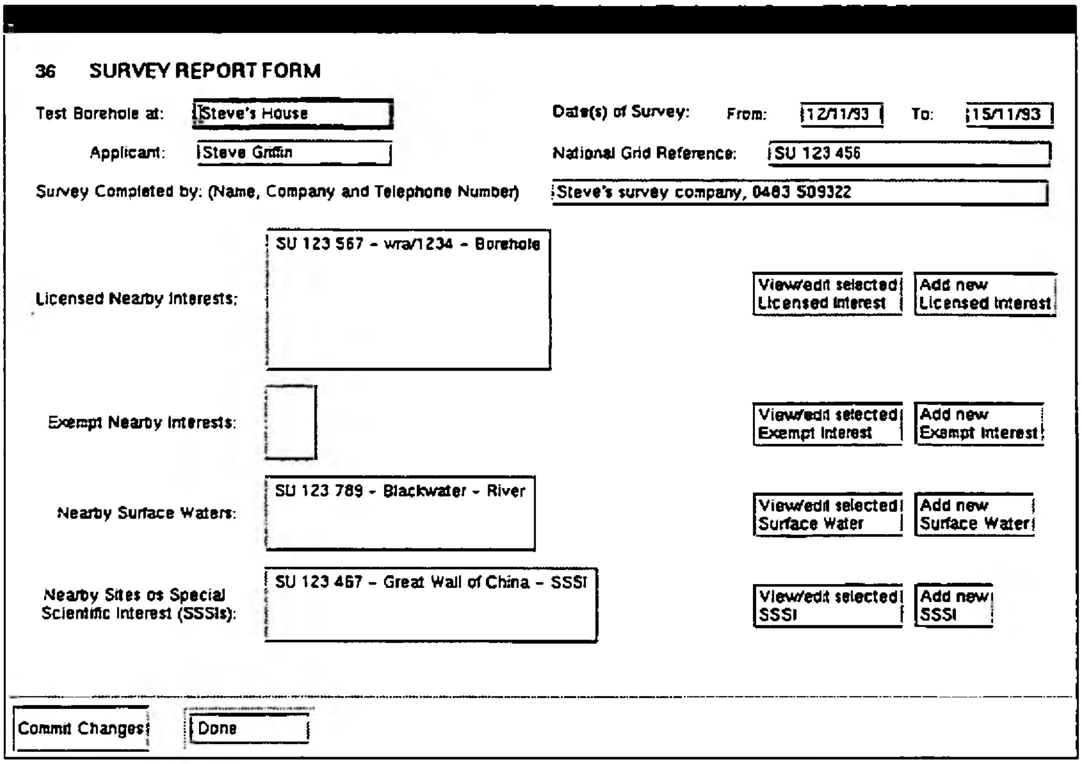


Figure 36: The Test Pumping Survey Report Form

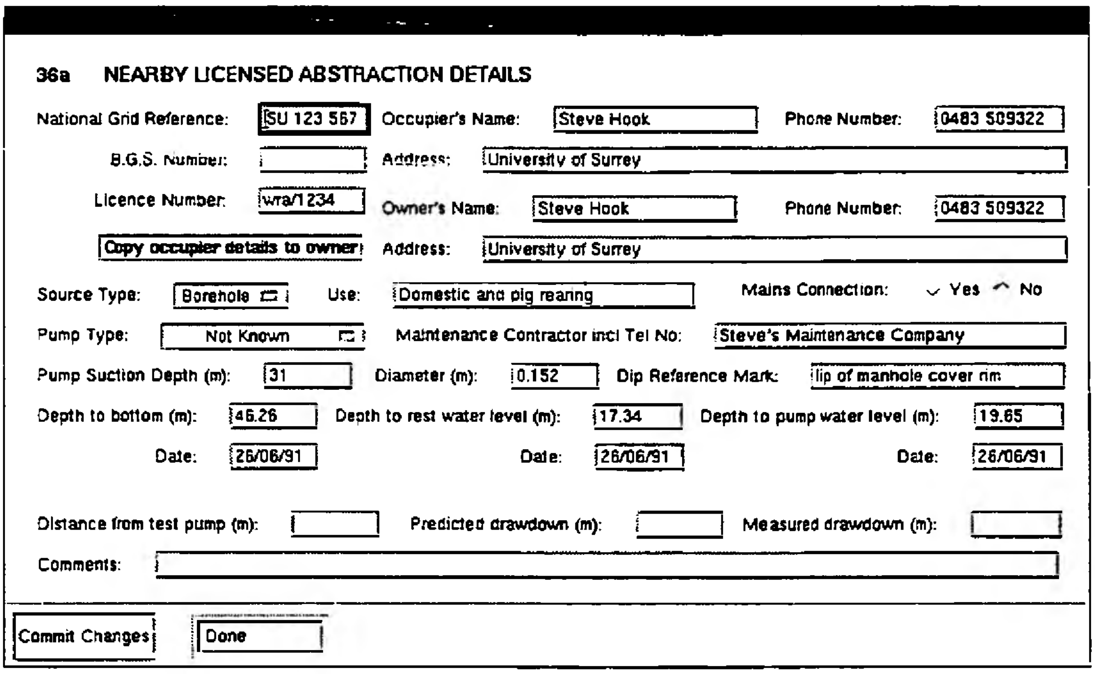


Figure 37: Details of Nearby Licensed Abstractions to the Proposed Borehole

The details, for nearby licensed abstractions for example, include a summary of the licence details, such as the licence number, the NGR, occupier, owner, the source type and water use, and details important to the current application, such as the distance from the proposed source, any measured or estimated drawdowns in the licensed source, and any other comments on the effect of the proposed abstraction. This information is used in the Groundwater Investigation task for test pumping data analysis, the results of which are used in the derogation task.

As stated earlier, many more screens can be seen in the ELSIE User Manual 406/2/S.

6. THE ELSIE LEGISLATION BROWSER

The Legislation Browser is part of the ELSIE system, however it has been developed as a stand-alone application on a PC. The Legislation Browser currently provides hypertext access to the 1991 Water Resources Act.

6.1 Accessing the Act through the Legislation Browser

There are a number of information retrieval (IR) programs, mainly used in libraries, that can retrieve documents using a keyword search. The essential pre-requisite for this operation is that the keywords have to be appended to the document which is to be searched and there are programs that match user queries with these keywords. The IR programs normally access the whole document, and where such programs access parts of a document, then each part has to be appended with the appropriate keywords. There is no doubt that this is a labour intensive process and there is always the possibility that, if the appendages are made by humans, some chunks would be missed and the danger of misspelling a keyword can be potentially quite hazardous.

The use of conventional information retrieval programs for searching and accessing aspects of the Water Resources Act 1991 is a possibility. However, we believe that the use of hypertext documentation, as described in Section 6.5 below, is ideal for the needs of licensing officers within the NRA, as well as many more NRA staff. Figure 38 shows Section 193 (2) of the Water Resources Act in the Legislation Browser.

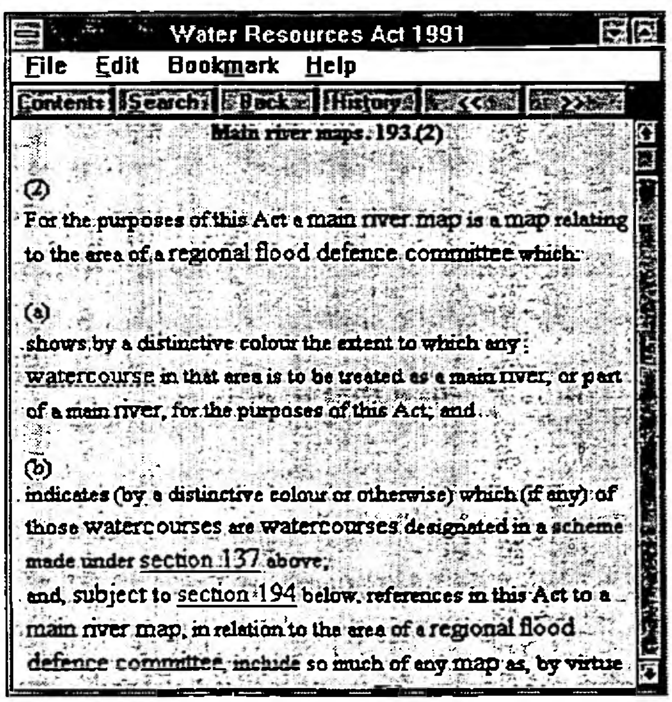


Figure 38: Legislation Browser showing Section 193(2) of the 1991 Water Resources Act

Each Section of the Act can be accessed by following a sequence of contents pages. By pressing the "Contents" button on the browser the user is given a full list of Parts of the Act. From here the user may select the required Part to get a list of the Chapters within it. This leads to a list of Sections of the selected Chapter from which the user can select the requested Section which is displayed as shown in Figure 38.

The "Back" button takes the user back through previously viewed Sections or contents pages sequentially, but if the user wishes to return to a Section viewed some time ago, the "History" button provides a list of all previously viewed pages for the user to select from.

If the user is not sure which Section of the Act he/she requires, a search may be conducted for a list of key words or phrases within the Act. For example, to access the Section illustrated in Figure 38 above, the user may request a search for all occurrences of the word "watercourse", as shown in Figure 39 below.

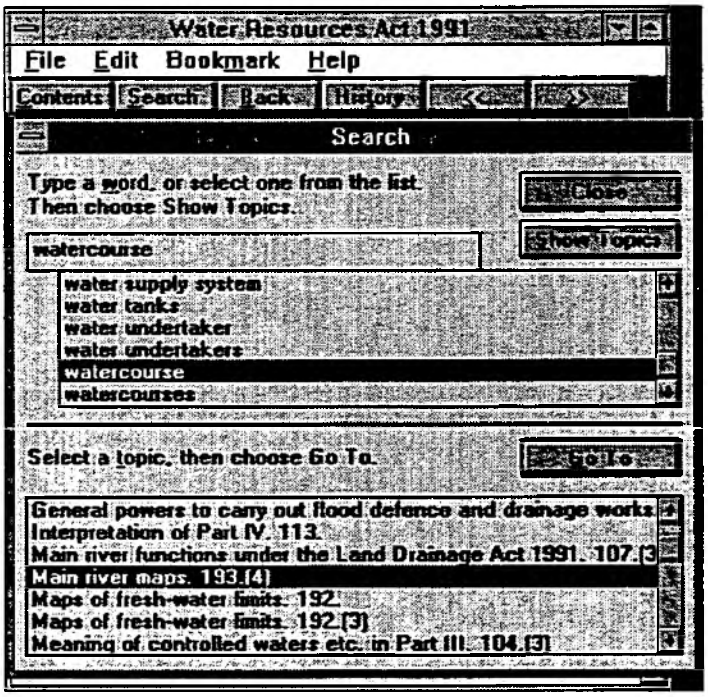


Figure 39: Searching for topics related to 'watercourse' in the 1991 Water Resources Act

By selecting the term "watercourse" and pressing the "Show Topics" button, a list of all Sections containing the term, and their titles, is supplied by the system in a scrollable box. The user may move to any of these topics by selecting his/her choice in the box and pressing the "Go To" button. If the Section selected is not the one required, the search window may be recalled and the previous search results will be retained in the box for another Section to be chosen.

Many Sections of the Act refer to other Sections, Chapters or Parts of the Act as we have discussed earlier. In Section 193(2), shown in Figure 38 above, we see reference to Sections 137 and 94. Each reference of this kind, throughout the Act, is highlighted to the user who, on selecting such a reference, is instantly taken to the highlighted section. This is a 'jump' as described in Section 6.4.2.

The final feature of hypertext documentation available through the Legislation Browser is the 'overlay' facility for terms defined in the Act. The interpretation Sections of the Act provide definitions for many terms, some relevant only within the context of certain Parts of the Act, some relevant the Act in its entirety. Such terms are also highlighted in the Legislation Browser, and also underlined with a dotted line to distinguish them from cross-reference jumps. On selecting a defined term the user is supplied with the term's definition in a box above the main Section text currently being browsed.

Figure 40 shows the definition of the term "watercourse", taken from the Interpretation Section of the Act, and displayed to the user in an overlay box.

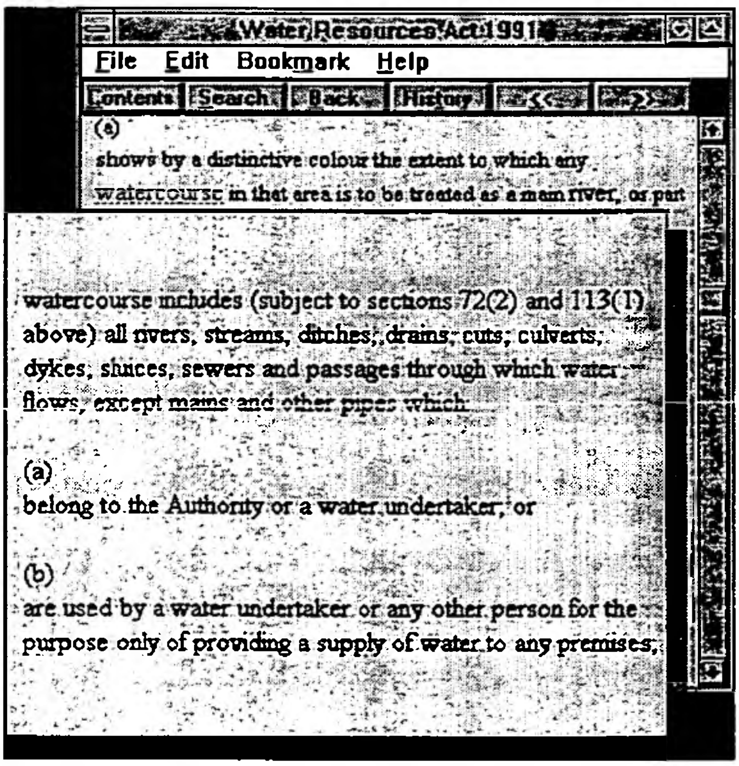


Figure 40: Definition of the term 'watercourse' as an overlay in the Legislation Browser

Two other features of the Legislation Browser valuable to an NRA operative are the bookmark and annotation facilities. Adding a bookmark to a frequently viewed topic allows the user to move quickly to that topic by pulling down a menu of bookmarks from the "Bookmark" option at the top of the browser and selecting from a list of his/her marked Sections. The annotation facility, found on the menu under the "Edit" option, allows the user to add his/her own comments to any

Section of the Act, currently achieved by scribbling in the margins of their paper version: their annotations can be viewed whenever the associated Section is being viewed.

6.2 The Water Resources Act 1991

The Water Resources Act (WRA) 1991 was enacted by the UK Parliament to establish, empower and regulate the functions of the National Rivers Authority.

The 1991 Act, which is a revision of the Water Resources Act 1963 and Water Act 1989 and whose origins can be traced back to the Water Act of 1945, is divided into nine Parts. Like many other UK Acts of Parliament, these nine Parts are subdivided into many chapters and a total of 225 Sections, followed by 26 Schedules. Table 15 shows the names of the nine Parts and the Sections in each Part. The day-to-day application of the Water Resources Act 1991 relies on constant look-up these sections.

Part	Title	Sections
1	Preliminary	1-18
2	Water Resources Management	19-81
3	Control of Pollution of Water Resources	82-104
4	Flood Defence	105-113
5	General Control of Fisheries	114-116
6	Financial Provisions in relation to the Authority	117-153
7	Land and Works Powers	154-186
8	Information Provisions	187-206
9	Miscellaneous and Supplemental	207-225

Table 15: Parts of the 1991 Water Resources Act

These Parts are cross-referenced with each other. The rows in Table 16 show how many times Parts of the WRA 1991 refer to other Parts. For instance, Part I refers to Parts 2 three times, Part 3 once, Part 4 twice and Part 5 four times. However, Part 5 is independent of all others and only refers to itself.

There are, at least at the Parts level, a total of 130 cross-references: Parts 2 (Water Resources Management) and 3 (Control of Pollution of Water Resources) are the most frequently referred to (40 and 39 times respectively), followed by Part 4 (Flood Defence - 21 times) and Part 7 (Land and Works Powers - 19 times). Now, if we look at which Part most frequently refers to others, we find Part 6 (Financial Provisions in relation to the Authority) has 26 references to other Parts (mostly Parts 2 and 3), followed by Part 7 (24 references). Part 3 most frequently refers to itself.

Part	Parts Referred To									Total
	1	2	3	4	5	6	7	8	9	
1	-	3	1	2	4	-	1	-	-	11
2	-	7	1	-	-	-	1	-	-	9
3	-	-	18	-	-	-	-	-	-	18
4	-	-	-	8	1	1	-	-	-	10
5	-	-	-	-	1	-	-	-	-	1
6	-	13	10	2	-	-	1	-	-	26
7	-	3	2	5	-	-	14	-	-	24
8	1	4	4	1	-	1	2	1	1	15
9	-	10	3	3	-	-	-	-	-	16
Total	1	40	39	21	6	2	19	1	1	130

Table 16: Cross-reference of the parts of the 1991 Water Resources Act

The WRA 1991 has been released by Her Majesty's Stationery Office (HMSO) as a 260-page document; a closely-typed document comprising over 100,000 words, printed in a two-column format: a main text column together with elaborations/commentaries in the second, parallel column.

6.3 The WRA 1991 and other Acts of the UK Parliament

The WRA 1991 is a part of the UK environmental legislation and, as such, therefore refers to, and depends upon, other environmental legislation. Table 17 contains a list of some of the Acts of Parliament that are cross-referenced in the WRA 1991.

life assurance act 1774
diseases of fish act 1937
prevention of oil pollution act 1971
salmon and freshwater fisheries act 1975
ancient monuments and archeological areas act 1979
planning and land act 1980
food and environment protection act 1985
new roads and street works act 1991

Table 17: Some of the Acts of parliament cross-referenced in the Water Resources Act 1991

There are, of course, a whole range of uses of the water: from recreation to irrigation; water supports the life cycle of almost all living organisms; water is a transportation medium; the governance of the aquatic environment involves central and local government organisations; the supply and distribution of water may be interlinked with the supply and distribution of other strategic resources. Indeed, the range of water uses covers almost all walks of life.

6.4 Kev Words and Phrases in the 1991 Act

The WRA 1991 is written in a domain-specific or subject-specific variant of English: English legal language. Texts written in this specialist language can be distinguished from general English texts by the frequency of nominal expressions, long sentences and the preponderance of complex prepositional phrases. The 'weirdness' of this special language can be used similarly to the special language of domain specific texts and interview transcripts, as described in Section 3.
A detailed description of the use of special words and phrases in legal texts in general, and in the 1991 Water Resources Act in particular, can be found in R&D Project Record 406/3/S.

6.5 Producing Hvpertext Documentation

As mentioned above, the Legislation Browser is provides the user with the 1991 Water Resources Act in hypertext form. Hypertext is a generic term used to cover a number of techniques to create and view multidimensional documents. These documents may be entered at many points and may be browsed in any order by interactively choosing words or key phrases as search parameters for the next text image to be viewed.

6.5.1 Features of a Hypertext Document

We believe the following features are desirable for any hypertext documentation. Equivalent features are available through many applications' on-line help utilities, particularly those applications running under Microsoft Windows. However, such on-line help is not available, as far as we know, based on legal texts.

First of all, hypertext documents must be divided into topics: small sections of text covering a particular subject. Topics would normally be smaller than chapters or even sections in a book or paper, but may be as long as the subsections of this report, for example. This will make the document much more readable, providing more specific information for the reader who does not wish to browse through irrelevant material.

The basic feature of a hypertext document should be the ability to move from one topic to another related topic without searching manually through the whole document to find it. This facility is an extension to the conventional use of such annotations as "turn to page 3 for further details", however here the user does not need to know where further information is: he/she is just taken to that (relevant) section by selecting such a phrase. For example, all Acts of Parliament have an interpretation section which provides a glossary defining each of the important terms used in the Act. The user should at least have the ability to "jump" from a topic where such a term is used, to the glossary where it is described.

A glossary could be very large, however, and in the example above only a small part of the section is of use to the user requiring the definition of only one term. In the case of a large glossary, a hypertext document should have the ability to show a (small) related topic to the user, but not actually move to the new topic: what is required is a window with the relevant information in placed above the current topic. This we call "topic overlaying". Figure 40 shows this feature with the interpretation of the term "watercourse" in the Water Resources Act 1991.

There are two remaining features we desire in a hypertext document. A list of keywords should be available to the user like an index. From here, as above, the user should be able to jump to any occurrence of the keyword. Also, the topics should be arranged into a sequence to enable the whole document to be browsed, topic by topic.

6.5.2 Marking-up a Hypertext Document

Our methodology for hypertext document production involves the marking-up of text with tags which define the features introduced in Section 6.5.1 above. The mark-up language we propose is simple enough for a knowledge engineer or domain expert to read, and therefore apply to the text manually if necessary, yet constrained enough for a computer program to translate into a form readable by a hypertext viewer. It has been designed so that even on paper the mark-up is a useful source of information.

We call our mark-up language ADML (Active Document Mark-up Language). By "active documents" we mean documents which are marked up in such a way as to capture the essential meaning of a given document. Details of ADML and how it was used to mark-up the 1991 Water Resources Act can be found in R&D Project Record 406/3/S.

7. ELSIE USER TRIALS

The User Trials were the most innovative aspect of the ELSIE project, particularly in view of the fact that this was an R&D project albeit with a strong applicative flavour.

The Trials were conducted at three sites within the NRA. In addition to the selected members of licensing staff at each of the Regions, the Trial was attended by licensing and R&D staff from other regions including the Headquarters at Bristol. Table 18 provides details of the ELSIE User Trial sites, dates and attendees (a complete list of attendees is attached in Appendix D).

NRA Region	Start Date	Finish Date	Other attendees
Southern	16 February 1994	23 February 1994	Headquarters: R&D directorate
Thames	28 March 1994	8 April 1994	Anglian Region, South Western Region, Southern Region.
Northumbria and Yorkshire	25 April 1994	3 May 1994	Post-project demonstration / trial is being arranged.

Table 18: Details of the ELSIE User Trials

Each User Trial began by a demonstration of the ELSIE system by the University of Surrey. The attendees were provided with the ELSIE User Manual and were asked to note potential and actual errors on an 'ELSIE User Testing Problem Report', as shown in Figure 41.

ELSIE User Testing Problem Report

Reported by: ______________________ Date: __________
Region: ______________________

Problem Area:
User Interface _
APMan _
WALDES _
Database _
Legal Browser _
Other (specify) _ ______________

Problem Description:

Figure 41: ELSIE User Testing Problem Report

The above figure shows that the attendees were asked to comment on five areas of ELSIE's operation together with an '*others*' slot. The attendees found the User Interface to be satisfactory, and despite the fact that the database interface of ELSIE, that is the interface between

ProKAPPA™ and ORACLE® (Version 6), is not quite as robust, only few problems (c. 10%) were related to database operations.

The overwhelming number of errors, just about 50%, were related to WALDES, followed by 34% for the APMan subsystem. A number of these errors were fixed, some of the errors were trivial, whilst others caused embarrassing failures of the ELSIE system in the presence of invited audience. Most of the errors could be traced back to the way ProKAPPA™ has been implemented by its vendors. Nevertheless, some of the errors relate to the knowledge in ELSIE's knowledge base. The attendees hoped that these errors, particularly the ProKAPPA™ related errors, could be fixed so that they could realise the true potential of the ELSIE system.

A number of attendees of the User Trial have expressed their appreciation of the Catchment Water Balance Tool (CWBT), a part of WALDES, and expressed a desire to use CWBT.

As we have indicated earlier, the Legislation Browser has been implemented on a PC and uses Microsoft's WinHelp program. The attendees showed their appreciation of the Legislation Browser and no negative comments were received.

Table 19 comprises a breakdown of the errors by the attendees on the ELSIE User Testing Problem Report.

Operational Aspect	Number of reported errors
User Interface	5
APMan	26
WALDES	41
Database	3
Legislation Browser	N.A.
Other	9
TOTAL	84

Table 19: Breakdown of errors by ELSIE component reported during the User Trial

Of the 84 errors reported the University of Surrey has fixed 30 so far and are unable to reproduce 13. Furthermore, 10 of the errors, caused entirely by the make-up of ProKAPPA, have been temporarily fixed. This leaves 34 errors which either need long term solutions or can be alleviated. The breakdown is given in Table 20.

The categories listed for each error were taken directly from the attendees' description and in some cases the attendees have identified errors with one operational aspect, say APMan, where in reality the error lied in another part of the system, like the database. However, such miscategorisation does not alter the overall trends in the above data.

Operational Aspect	Fixed	Cannot reproduce	Temporarily Fixed	Pending	TOTAL
User Interface	5	-	-	-	5
APMan	9	7	5	5	26
WALDES	15	4	5	17	41
Database	1	-	-	2	3
Legislation Browser	-	-	-	N.A.	-
Other	-	-	-	9	9
TOTAL	30	11	10	33	84

Table 20: Breakdown of errors by ELSIE component and status of fix

7.1 User Interface

The user interface of most software systems leaves much to be desired. The design of a user interface requires an in-depth understanding of the organisation for which the software has been designed together with a detailed knowledge of potential users and their psychology-at-work. In order to address these crucial and sometimes insoluble problems it was decided by the University of Surrey to use the various abstraction licensing application forms as the *user interface*. The ELSIE system's user interface is therefore 'familiar' to its potential user in that the forms were developed within the NRA. The design of these forms would have required an understanding of business at hand, that is abstraction licensing, and some psychology-at-work of the licensing officer. Forms were displayed by ELSIE as true a facsimile as possible, complete with the colour of the paper on which the forms were printed. The entirety of ELSIE's operation is driven by the slots and boxes one is meant to fill in on these forms.

The user interface part of ELSIE's operation attracted no unfavourable comment during the trials. Indeed, out of the 84 errors reported by the Trials at Southern and Northumbria & Yorkshire only five were attributed by the attendees to the user interface. Out of these five, at least three are related to other aspects of the system.

7.2 Applications Progress Manager (APMan)

The APMan subsystem is of potential import in the management of an abstraction licence application. The term management refers here to the management of the entire lifecycle of the application. This includes the original submission of the application right through its acceptance or rejection, and finally to its retrieval for comparing the (pre-stored) application with a new one. There is a time lapse of three months in the processing of an application, and much longer time elapses preparing to make the application in the first place. APMan keeps a diary of the progress throughout this process and makes the diary accessible by other NRA operatives.

The development of APMan required us to establish an interface, through proprietary software, between the ProKAPPA™ programs that comprise ELSIE and ORACLE®, a relational data base management system. Such an interface is an interface of two distinct technologies: knowledge based systems and data base management systems.

The abstraction licenses cover many different purposes, crops, etc. and require correct topographical details of source of water. The ELSIE knowledge base stores a substantial number of details of these purposes, crops etc. However, as the uses of water are diverse in the extreme, APMan sometimes fails to recognise, for example, that *carrots* could be *spray irrigated* or that the users would need to add more crops to APMan's data base. (The latter, of course, should be controlled and only a system's administrator should add or delete from APMan's data base.)

The breakdown of errors in APMan are listed in Table 21.

Error Type	Number	Comments
Add new categories	10	Crops, Source Map Reference
Data Conflict	5	Dates not allowed
Application Storage and Retrieval	5	Cannot retrieve or store applications
Miscellaneous	6	Wrong default answer; Interaction between APMan and WALDES
TOTAL	26	

Table 21: Breakdown of errors in APMan

A number of the above errors have been fixed: nine permanently and five had a temporary fix put in. Five errors could not be reproduced while the knowledge engineer was present and five errors are yet to be fixed. It is hoped that APMan is now much more stable than it was during the User Trials.

7.3 WALDES

The testing of WALDES has contributed to the greater proportion of problems and errors in the operation of ELSIE. A good number (c. 30%) of the errors were serious in that the system simply crashed and the user had to re-start: most of these errors were related to object management within ProKAPPA™. Some of these errors were fixed by the University of Surrey and others have been reported to the vendors of ProKAPPA™: IntelliCorp® Inc.

Like APMan, there were a number of errors also due to a 'lack' of knowledge or 'incorrect' knowledge in the ELSIE system. These account for 10 out of 41 errors reported for WALDES (c. 25%). However, it is worth noting that some of the problems highlighted by the User Trials here have evolved during system develoment: for example, during knowledge acquisition all parties were happy with the fact that the 'Water Quality Board' should be consulted, however now, some regions reported that no such Board exists.

Table 22 shows the breakdown of errors reported for WALDES.

Error Type	Number	Comments
Add new categories	10	Navigation; some NRA regions do not have Conservation Authority. etc;
Object and Memory Management in ProKAPPA	13	'Empty' objects leftover from a previous run
Slowness of operation and related errors	9	System takes five minutes or more in some cases
Miscellaneous	9	Catchment Water Balance tool problems; Interaction between APMan and WALDES
TOTAL	41	

Table 22: Breakdown of errors in WALDES

The University of Surrey has fixed 15 of the 41 reported errors and provided a temporary solution to a further five. We could not reproduce four of the reported errors and 17 of these 41 errors were classified as requiring a longer term solution.

7.4. Database and Other Operational Aspects of ELSIE

The problems related to the database aspect of ELSIE's operation were small as compared to, say, APMan and WALDES. A total of three errors were reported, one was fixed and the remaining two are minor in nature. What is required here is a more robust interface between ProKAPPA™ and ORACLE®. It has to be added here that the modelling of data within ORACLE and the query language required to access the data is a lower level than, say, data modelling techniques and access in ProKAPPA, and consequently requires a different kind of expertise than was available within ELSIE. However, specialist programmers were hired to address this problem and it is hoped that most of the problems in this area have been overcome.

The important point to note here is that ELSIE can be interfaced to a *proprietary* data base management system. This means that, if and when the NRA decides on an appropriate data base system for, say, National Licensing Data Base, ELSIE can be easily interfaced to that system. If the choice was ORACLE then there would be no effort required, but even if the choice was any other relational data base system the effort required would be minimal.

The 'other' category of errors usually required a request for an added feature in ELSIE. Comments on drawdown calculations and on the Catchment Water Balance tool were also incorporated in the 'other' category.

7.5 Conclusion

The User Trials undertaken for ELSIE were a success. This was, perhaps, the first time that a software system that originated from an R&D project was tested in a 'live' situation with the software development team to hand. Furthermore, the Trial was on an almost National basis in that it was conducted at three locations (Reading, Worthing and York) and the evaluation input was received from four out of the ten regions of the NRA.

The overall impression of the ELSIE system was favourable. Southern Region noted that 'all the ELSIE project objectives have been achieved, some more successfully than others. Some tailoring of the system will need to be done for each region, e.g. inclusion of underground strata as appropriate, river names etc.'. The evaluators at Northumbria and Yorkshire said: 'The system has great potential as a tool in the licensing determination process. It would speed training as an additional element which would help, but not replace, conventional methods providing a great insight into the licensing process'.

The User Trial results at Thames NRA found that the 'Legislation Browser attracted most interest', followed by APMan in that APMan can check the content of the application forms during the input phase. Their comments were guarded as far as WALDES was concerned, as was the case to a greater or lesser degree at other User Trial sites, as WALDES accounted for a large number of errors during the ELSIE trial.

The biggest drawback of the ELSIE system in its present form are the system level (ProKAPPA and ORACLE) errors that the attendees at the Trial encountered. As the attendees at Thames Region, that included visitors from South Western and Southern Regions, remarked that 'WALDES suffered from the "Empty Objects" problem [a peculiar ProKAPPA induced error] throughout the Trial and never performed to a standard which could be expected to impress'.

Almost without exception all the User Trial sites reported a preference for a PC-based hardware platform, running under Windows, to a SUN-based hardware platform running under UNIX.

The entire user reports from each of the three regions can be found in R&D Project Record 406/3/S.

8. CONCLUSIONS

The ELSIE project has demonstrated the effectiveness of advanced information processing systems in determining abstraction licences. The ELSIE system comprises two major components: the intelligent advisor, that helps in evaluating an abstraction licence and keeps track of the application's progress, and the legislation browsing component that helps in searching a legal data base that currently contains the Water Resources Act 1991.

8.1 Abstraction License Determination as an Area for Building Information Systems

Abstraction licensing requires a subtle balance of environmental, engineering and legal considerations: expert systems are built usually to encapsulate and disseminate the knowledge of an expert in a very narrow field of specialisation, thus typically a robust system can be produced for just the environmental or engineering or legal knowledge dissemination. These narrowly based systems do not really simulate the expert's behaviour in a way expected by novices. The progress in expert systems development environments, in text-based knowledge acquisition, and in brainstorming, has encouraged the knowledge engineering community to take on projects like ELSIE.

The involvement of the project steering group and their continual interaction with the ALG resulted in significant broadening of the scope of the project. However, this involvement also meant that the contractors were able to determine the needs of the licensing officers who actually determine licence applications rather than build a system on a hypothetical need. Without this ambitious approach, ELSIE could not have become the focus of debate within the ALG which involves those who think they require the system as it is whilst others think that it is too ambitious.

Four major areas in licence determination were chosen: licensing of groundwater sources; licensing of surface water sources; impoundments; and legislation and licensing procedures. Again, it is easier to build a very robust expert system for one of the four areas within the time-scale of the ELSIE Project. But such a narrowly based expert system would not serve the needs of the licensing community, and more to the point, such a narrowly focused system cannot be evaluated as effectively as the more broad based system like ELSIE.

8.2 Knowledge Acquisition and System Specification

ELSIE was developed with the active involvement of licensing officers from four NRA regions, namely Thames, Wessex, Southern and Yorkshire. These officers selected experts from Thames, Wessex, Welsh and Severn-Trent who were subsequently interviewed, on topics selected by the licensing officers, for the purposes of eliciting the knowledge of the experts. This knowledge forms the basis of ELSIE's operation and is used by ELSIE to guide a licensing officer to determine an application for an abstraction licence.

8.3 Software Development

The software for the ELSIE system was developed at the University of Surrey and was tested throughout the development life-cycle. The advisory components, APMan and WALDES, were developed on a SUN-SPARCSTATION using ProKAPPA™, a large-scale software development environment. The APMan component, that helps in filling an abstraction licence application, has been interfaced to an ORACLE® data base such that an application, either completed or in the process of completion, can be stored in an applications database. This means that when the NRA national databases for abstraction licenses are on stream, then ELSIE data can be either passed on directly to the database, if the national database is in ORACLE, or with a relatively small effort ELSIE can be interfaced to other proprietary databases, provided KAPPA can establish communication with the chosen software.

There are over 200 tasks identified through the interviews and subsequent analysis. Typically, each task spawns between 10 to 100 rules which means in principle there can be as many as 2000 to 20000 rules in the system. The design of ELSIE, particularly its user interface which is form orientated in that NRA forms were used as the interface, has relieved the need to write a rule for every possible input option. Therefore, each completed task has much fewer than ten rules associated with it, and some tasks have been subsumed by parent tasks completely, hence the current rule count is only about 200. If every task is analysed in complete detail, the rule count will go up: currently 76 tasks have been dealt with in some detail. We must stress that our user interface design strategy has allowed us to code in fewer rules since we represent many as option-menus and dynamic links between forms.

The legislation browser was implemented on a PC-system and uses a Microsoft®-supported hypertext system to store properly indexed legislative documents and navigate users in searching through these documents. The legislation browser is ready for day to day use and can be used in conjunction with Microsoft Word™ for hypertext term lookup.

8.4 User testing

During the project extension phase, December 1993 - June 1994, the system was tested by the potential users in various regions in situ, that is in their offices or in very close proximity of their offices. These regions included Southern, Thames and Yorkshire. Such testing regime is novel and gives the potential user a feel for the system in an environment which cannot be provided within the, rather artificial, environment of a software development laboratory. The results of the testing phase have been very encouraging and the feedback, critical at times and valedictory at others, has been on the whole quite positive. Those involved in testing have documented their comments and have written on the costs and benefits of the ELSIE system (see R&D Project Record 406/3/S).

8.5 Running ELSIE within the NRA

Legislation Browser: The use of this system will depend upon the NRA having an organisation wide license for keeping the HMSO produced electronic version of the Water Resources Act 1991. Typical single user license for the Act is £700.00, but a site licence can be negotiated by the NRA at a much lower cost.

The browser is currently implemented to browse using the Microsoft WinHelp System. It would therefore be essential to have Microsoft Windows™ licenses to be able to use the legislation browser.

Expert System: The expert system can be implemented within the NRA provided the NRA either has run-time licences or the NRA has a full development licence. The possession of the run-time licences means that the NRA would be able to use the system for day-to-day use but would not be able to modify the knowledge or change anything in the programs. The full licence will enable well-trained knowledge engineers to modify and maintain the system. The difference in the cost of the two licences is quite sharp: £1500 for the run-time and £15,000 for the full licence, or £350 for a PC run-time if the system's user interface was modified to enable the port.

9. RECOMMENDATIONS AND IMPLICATIONS FOR THE NRA

Although the current system works to the extent that all R&D objectives have been met, the Contractors would like the NRA to develop the ELSIE information environment such that the ELSIE system is available as a desk-top decision making system throughout the NRA.

It is in the nature of expert systems that their knowledge bases have to be almost continually updated, modified and, in some cases, extensively pruned. Indeed, it is in the nature of an experts' expertise that whenever a knowledge base is assembled, which in itself is a non-trivial task, there is a need to refine, prune and/or extend it. ELSIE is no exception to this. Section 9.1 outlines the state of the knowledge bases within ELSIE and spells out a strategy to improve the current state and make the system really worth while for the NRA to use nationally in the future.

The continual and rapid technological advances in hardware platforms and software systems, particularly in the context of expert systems, means that a program can be generally twice as faster on a hardware/software platform that costs only half as much two years ago. One of the primary goals for such a wide spread usage of the ELSIE environment necessitates the migration of ELSIE from its current expensive and eclectic hardware/software configuration onto a cheaper and widely available technology base. There are three major points to consider here: hardware/operating systems platform; implementation software; and, the user interface. (This is discussed in Section 9.2). Section 9.3 outlines a suggested implementation plan.

9.1 Knowledge Requirements of ELSIE

The user trials have shown that The ELSIE Information Environment can be of substantial value for abstraction licensing work. There are two potential areas of development that have to be considered for the NRA-wide desk-top oriented use of the ELSIE system:

1. LONG TERM:	Knowledge documentation and dissemination.
2. SHORT TERM:	(a) Legislation and guideline browsing (b) Office Management (c) Quick calculation tool

The outstanding knowledge acquisition tasks associated with the further refinement of ELSIE's knowledge base are shown in Table 23 below.

TASK	STATUS	WORK REQUIRED
Knowledge Acquisition	Baseline knowledge acquired	Revisions Expected
ELSIE Components		
a. Applications Progress Manager	User Interface finished; Links with external data bases established	Some debugging required; Links with NALD and WAMS
b. Water Abstraction Licence Determination Expert System	User Interface finished; Links with external data bases established	Limited knowledge utilisation; Extensive development required
c. Legislation Browser	Water Resources Act 1991 fully marked; system operational	Minor bug fixing; Copyright clearance from HMSO

Table 23: Outstanding knowledge acquisition tasks

Effort Required to build a fully operational ELSIE environment is estimated to be about 7 person-years over a two year period. A modular approach is recommended which involves the sequential development of the various knowledge bases as outlined in Table 24.

Module	Description	Start	Finish	Human Resources
APMAN	Licence Manager	Month 1	Month 12	2
WALDES I	Licensing Expert	Month 12	Month 18	2
WALDES II	Extended and Linked Version	Month 18	Month 24	2.5
Legislation Browser II	Land Drainage Act	Month 1	Month 6	0.25
Legislation Browser III	EC Environment Directive	Month 6	Month 12	0.25

Table 24: Recommended further development of ELSIE

However, before a discussion of this table, it is important to note a number of Information Management Initiatives that are currently either planned or, perhaps, are in the early stages of the process of execution. These include the development of the National License Data Base and the National License Manual.

It has to be remembered that an organisation as complex as the NRA is continuously evaluating new technology and finding new methods and techniques that may improve its overall performance. The development of ELSIE is a part of this find-evaluate-use strategy. The development of ELSIE's knowledge bases was an important first step in the computer-based documentation of abstraction licensing expertise. In this respect this development has complimented the deliberations of the ALG and other groups within the NRA.

The fact that a National License Manual is being developed is to be welcomed but there will always be a need to interpret the Manual: a Manual is not the description of how its content should be operationalised. ELSIE's knowledge base contains that extra knowledge, and have the capability of incorporating this knowledge for interpreting the Manual.

The NALD again is a data base and provided the design of the data base is professionally drawn and the user requirements of such a data base are exhaustively tested, there is no reason why such a data base cannot be interfaced to the ELSIE knowledge bases. Indeed, there is an argument here to suggest that the NALD designers' should have a look at the ELSIE's design of a licensing data base, in that ELSIE's design has been operationalised, whereas the NALD initiative is still evolving.

9.2 Hardware and Software Requirements of ELSIE

In the first instance, there is a requirement to move from the UNIX-based SUN SPARCSTATION (c. £5, 000) to a WINDOWS-based PC system (c. £2,500). The price-performance, that is the amount of computing power for a fixed sum of money, of the UNIX-based systems would always be greater than the comparable WINDOWS-based PC system: there maybe substantial problems in linking to large data bases and in the execution of large programs (c. 16 MB plus) on the PC-based systems. Nevertheless, the continual fall in the price of PC-based systems and the enhancement of the hardware power available, together with establishment of distributed data bases, the price-performance factor will, in the next two years, not be a major problem. The over-riding advantage, in our view, of using a PC-based systems is that the human resources required for maintaining such systems are more widely available and usually cost only half of what is required for the UNIX systems.

The second requirement is to migrate parts of ELSIE from the current expert system development environment Pro-KAPPA into a programming language, like C++ or Visual Basic. Pro-KAPPA was crucial to the development of ELSIE, particularly the way in which this software system supports the debugging of complex and interconnected knowledge bases, like the ones found in ELSIE. Pro-KAPPA is an expensive software system (c.£5,000-£10,000) and, like UNIX based systems, the human resources required to support programming and maintenance in Pro-KAPPA is not widely available and costs substantially more than what one would pay programmers' well-versed in C++. Moreover, Pro-KAPPA is a large and complex program which itself is evolving and is not quite a mature enough product that can be supported by an in-house team of programmers who may have other calls on their time.

In the near term, as Pro-KAPPA run-time programs, that is programs that do not need development or bug-fixing and hence can be run without the need of the full Pro-KAPPA environment, it would be possible to take a run-time image of ELSIE's knowledge base and be able to use that program on a UNIX-based system. It appears that Pro-KAPPA vendors have made it possible to execute the run-time versions on a PC-platform. This would enable us to achieve the 'desk top' objective of making ELSIE available throughout the NRA. However, the problem here is this that if and when ELSIE needs to be updated, somebody or some organisation will have to re-compile the knowledge base on a UNIX-SUN SPARCSTATION platform and re-distribute the copies of the ELSIE information environment. The long term solution is to re-write the ELSIE knowledge base either on low-cost expert system shell, like KAPPA-PC™, which will mean that updating the knowledge-base will still depend on the availability/goodwill of KAPPA-PC™ programmer, or through the use of a popular programming language, like C++, and popular data-base management systems. The positive aspect of translating the knowledge base into C++ is that it is possible, according to Pro-KAPPA vendors, to convert the existing ELSIE knowledge base into C++ program code. The negative aspect of this conversion is that such a translation does produce very large programs: debugging large programs is not trivial.

The third point relates to the user interface. The ELSIE project has successfully overcome one of the key problems in knowledge engineering, that is user interface design and implementation by adopting the NRA-issued forms as the interface. This has helped the knowledge engineer to build a user-interface that uses specialist terms, and questions that use those terms, which are exactly the same as the keywords and questions used in the forms. The forms being the interface of the NRA to the external world have enabled the knowledge engineer to program ELSIE's request for input data in a language that is already in use within and outside the NRA. The forms serve as the basis of the interface and the intelligent use of dialogue boxes, windows and other icons has resulted in a robust user interface. Currently, the user interface is written in UNIX-based and is Pro-KAPPA oriented. The problems of cost and human resources availability, as encountered above, do concern us here also. Given that the target ELSIE system, the desk top decision support system, is to be made available on a PC, and there have been exciting new and low-cost developments in the area of user-interface, including the easy-to-use software systems that enable a relatively less experienced programmer the use of dialogue boxes, windows, radio-buttons etc., which might be of relevance here. We recommend that user-interface programs should be ported onto Visual Basic.

Figure 42 compares what is available at the present moment (July 1994) and what is possible in the near-term (c. October 1995):

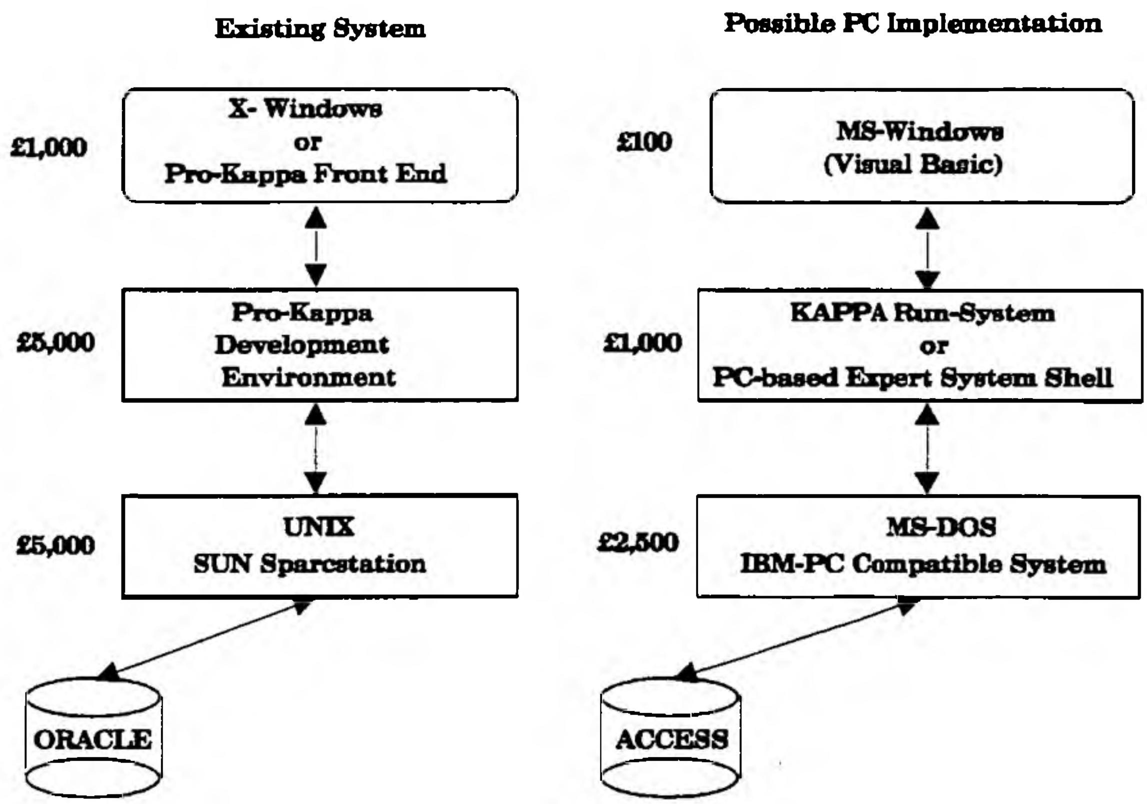


Figure 42: Comparison of present configuration of ELSIE and what is desirable in the near term

9.3 A Suggested Implementation Plan

We recommend a three stage implementation of ELSIE - entirely on the PC platform under Microsoft Windows. We suggest that an Implementation Steering Group be set up for the process along the lines of the ELSIE Project Steering Group but with more input from NRA IS staff. The three phases are outlined below with estimated resource requirements in person months.

9.3.1 Legislation Browser

The user trials suggest that the prototype of the legislation browser is almost deliverable as an implemented system. There are a few changes to be made, such as font standardisation between all topics, overlays etc., and more sophisticated indexing may help usability, however these are not large tasks. We also recommend the addition of the Land Drainage Act and the EC Environment Directive. As with the prototype, we recommend this system be delivered separately to APMan and WALDES, in the Windows 3.1 Help Format.

Estimated resources for implementation of Legislation Browser: 6 person month

We further recommend, however, that the NRA consider further development of the legislation browser to include other legal texts relevant to the work of the licensing officer and past cases where the legislation has been cited. Note that there is already a draft of the new Act for the "Environmental Agency" which will replace the 1991 Water Resources Act, and the NRA should consider how such replacements could be added to the browser.

9.3.2 APMan and the ELSIE Database

We recommend that APMan is taken out of the KAPPA™ development environment for implementation on a PC platform. The User Trials have suggested that APMan may be very useful to Licensing departments even without WALDES, and as a stand-alone system it does not require the functionality provided by such an environment. We suggest that APMan be rewritten as a client-server database system in a Windows development environment, such as Visual Basic or Visual C++. The format and functionality of APMan should remain the same, however data input and retrieval to the system will be greatly simplified and speed greatly improved by this approach.

The interface between APMan and WALDES would then be a loose-coupled type through the database: APMan would be used to input the data to be stored in the database which would then be retrieved by WALDES when and if necessary. Similarly, results from WALDES would be added to the database for APMan to relay to the user who has no need to start up the expert system. Some tighter coupling is foreseen to enable WALDES to use APMan front-ends for input of missing or incomplete data (see WALDES version 2 below), however this would be handled by the WALDES component.

The resources required here would mainly be on form production (a time-consuming but not difficult task), however careful consideration should be given to the design and development of the underlying database which will become the key component of the ELSIE package. This database must take into consideration, even at this early stage, the requirements of WALDES and NALD, to which it may one-day need to send data to for new licences and retrieve data from on existing licenses.

Although this approach means none of the system code of the existing APMan prototype may be used in implementation, the prototype makes an excellent 'executable' specification for the new version. Also, the steering group can ensure, at this early stage, that all foreseeable future requirements may be handled by the ELSIE suite.

Estimated resources for implementation of APMan: 24 person months.

9.3.3 WALDES (version 1)

The size and complexity of WALDES means that it will always need the sophisticated functionality of a large software development environment such as KAPPA™. However, this no longer requires an expensive UNIX platform to run the developed system. We recommend that WALDES remains a KAPPA™ (3) application, developed on SUN Workstations but delivered on the PC platform.

The user trials showed that, although potentially extremely useful to the Licensing officer, particularly a novice, the prototype WALDES is not yet ready for implementation as it stands. Three main areas need further development during the implementation phase. Firstly, the task base currently implemented could be expanded: it has been agreed that the current version covers a broad enough area as is necessary for R&D purposes, but more should be added for an implemented National system. Secondly, the database developed under the research project, is not suitable for the Object Oriented structure used for internal storage of data within WALDES. Finally, the user interface, developed in ProKAPPA™ 2.0 can not be ported to the PC platform: this means that each interface component must be re-engineered in KAPPA™ 3 prior to a port.

The problem with the scope of the task base will be forever ongoing: this is a classic problem with expert system development, such a system can always do more. We suggest that the steering group, after consultation with prototype testers and other licensing staff throughout the country, agree upon a scope at which point the first version of the system may be delivered. This first version, we believe, should be restricted to ensure that a version is implemented which is complete enough to be used by all regions but intended to be extended in the future.

The database problem is mainly due to it's structure and the database interfacing capabilities of the KAPPA™ system. The structure issue should be addressed during the implementation of APMan, and should therefore not affect the resources required here. The interface to the new database, however, must be redesigned and rewritten with the goals of speed and efficiency given priority. To this end, we recommend that an ODBC interface be created, which would allow fast and efficient data storage and retrieval to numerous database systems on any number of platforms, even over a network.

The issue of rewriting the user interface components in a format suitable for porting to a PC platform is, as with APMan implementation, more of a time consuming task than a difficult one. However, there is an added advantage here in that KAPPA™ 3 user interfaces have a very powerful linkage tool to the underlying system objects which is only partially handled by the home-grown linking functions which exist in the prototype.

Estimated resources for WALDES (version 1) implementation: 24 person months

9.3.4 WALDES (version 2)

We recommend that the second version of WALDES links the system to the implemented APMan and Legislation Browser, and the task base is considered for further development.

This will allow WALDES to use the efficient database handling capabilities of APMan to input missing or incomplete data and call on the Legislation Browser where useful to aid in the explanatory function of the system.

The task base development should at this stage attempt to incorporate as much scope as required by all regions of the NRA to make the system completely operational. It should be noted that this, however, will require further knowledge acquisition and a great deal of co-operation between future ELSIE users.

Estimated resources for WALDES (version 2) implementation: 30 person months

REFERENCES

Ahmad K. and Griffin S.M., (1991) "Expert Systems for Water Resources Management". Research and Development Report 3, London: National Rivers Authority.

Ahmad, K., Griffin, S.M. and Shaw, P. (1993) 'A Systematic Knowledge-based Approach to Licensing Water Resources'. In (Ed.). Coulbeck B., *Proceedings of an International Conference on Computer Applications for Water Supply and Distribution*, Leicester: De Montfort University (In press).

Bhatia, V.K. (1993) *Analysing Genre: Language use in professional settings*. London and New York: Longman Publishing.

Cruse, D.A. (1986) *Lexical Semantics*. Cambridge: Cambridge University Press.

Cruse, D.A. (1992) 'Antonymy Revisited: Some thoughts on the relationship between words and concepts'. In (Eds.) Adrienne Lehrer and Eva Fedder Kittay. *Frames, Fields and Contrasts: New Essays in Semantic and Lexical Organisation*. Hillsdale (New Jersey): Lawrence Erlbaum Assoc. Publisher pp289-306.

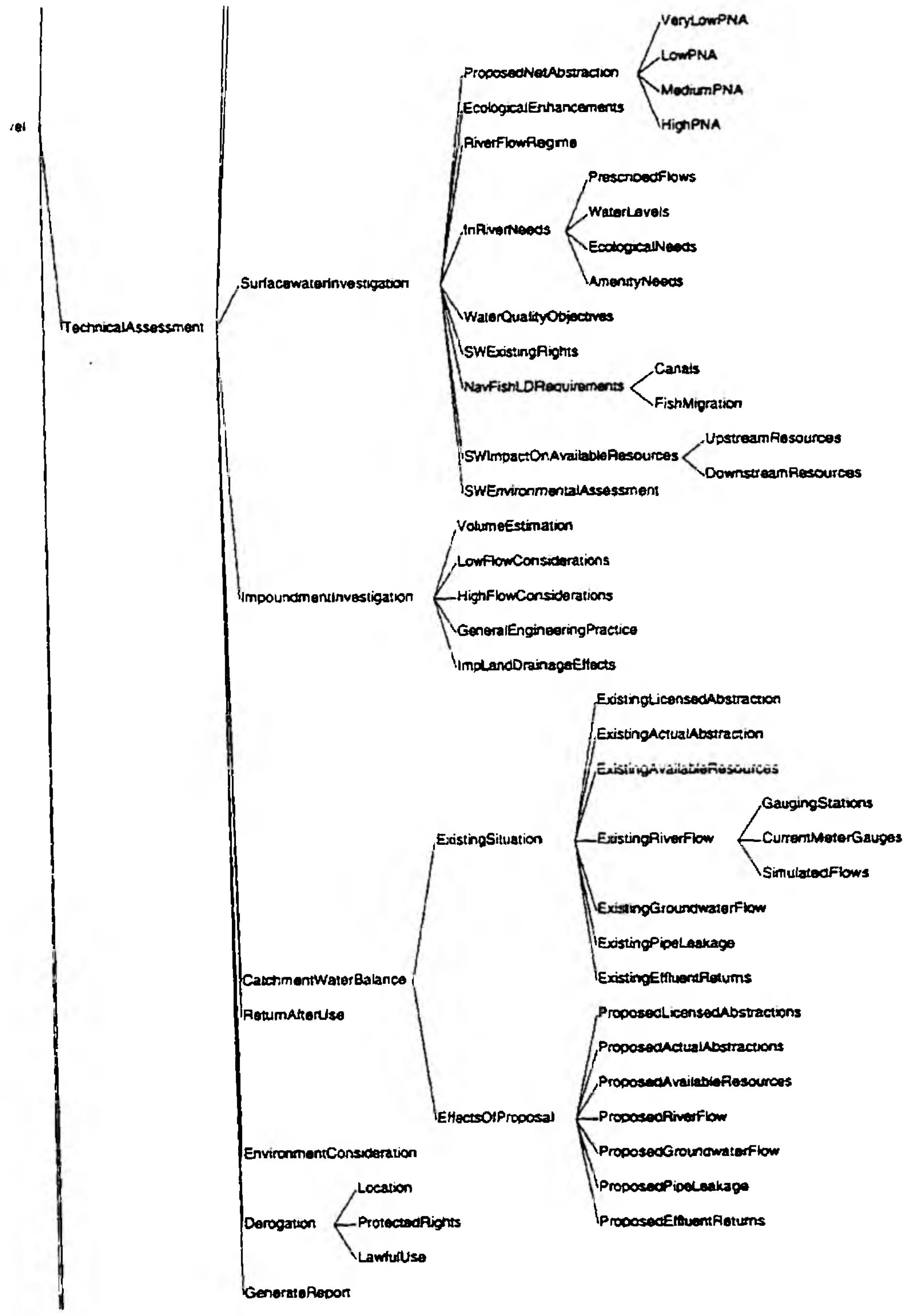
Hofland, K. and Johansson, S. (1982) *Word Frequencies in British and American English*. Bergen: Norwegian Computing Centre for the Humanities.

Jager, J.C. (1990). *Terminology Processing*. Amsterdam: John Benjamins

Picht, H. and Draskau, J. (1985) *Terminology: An Introduction*. Guildford: University of Surrey.

Quirk, R., Greenbaum, S., Leech, G. and Svartvik, J. (1985) *A Comprehensive Grammar of the English Language*. Harlow (Essex): Longman Group Ltd.

APPENDIX A: THE ELSIE TASK STRUCTURE


vel
TechnicalAssessment
SurfacewaterInvestigation
ProposedNetAbstraction
VeryLowPNA
LowPNA
MediumPNA
HighPNA
EcologicalEnhancements
RiverFlowRegime
InRiverNeeds
PrescribedFlows
WaterLevels
EcologicalNeeds
AmenityNeeds
WaterQualityObjectives
SWExistingRights
NavFishLDRequirements
Canals
FishMigration
SWImpactOnAvailableResources
UpstreamResources
DownstreamResources
SWEnvironmentalAssessment
ImpoundmentInvestigation
VolumeEstimation
LowFlowConsiderations
HighFlowConsiderations
GeneralEngineeringPractice
ImpLandDrainageEffects
CatchmentWaterBalance
ExistingSituation
ExistingLicensedAbstraction
ExistingActualAbstraction
ExistingAvailableResources
ExistingRiverFlow
GaugingStations
CurrentMeterGauges
SimulatedFlows
ExistingGroundwaterFlow
ExistingPipeLeakage
ExistingEffluentReturns
EffectsOfProposal
ProposedLicensedAbstractions
ProposedActualAbstractions
ProposedAvailableResources
ProposedRiverFlow
ProposedGroundwaterFlow
ProposedPipeLeakage
ProposedEffluentReturns
ReturnAfterUse
EnvironmentConsideration
Derogation
Location
ProtectedRights
LawfulUse
GenerateReport

APPENDIX B: EXAMPLES FROM THE PAPER KNOWLEDGE BASE

The following are a few of the paper rules elicited from interview 1: How to license groundwater abstractions.

Rule-4
If looking at the effects of pumping
and we are trying to make a decision on whether or not to allow a new proposal
then processes which cause fluctuations in water level need to be taken into account.

Rule-5
Processes which cause fluctuations in water level are pumping of water
and the way aquifers respond to changes to recharge and discharge
and changes in barometric pressure
and changes the sea and estuaries due to tides
and changes in river level due to the changes in flow in the river.

Rule-8
If the private domestic abstractions should exceed 20 cubic metres a day
then these abstractions do become licensable.

Rule-33
The information that the NRA requires about the results of the pumping test consists of
details of the construction of the borehole
and details of the strata that is penetrated
and the results from the water level observations
and data of the river flow measurements
and data of the spring discharges
and data about anything that has been specified in the Consent.

Rule-36
Looking at the available resource it should be taken into account
what the water is going to be used for
and how much of it is going to be consumed.

Rule-37
If the effluent is coming back to the resource system above
or very close to the point of abstraction
then a process of re-circulation of the resource can be utilised.

Rule-39
If the borehole is very close to the river
then water will be sucked out of the river and into the aquifer
because of the reversal of gradient from the river (induced recharge).

Rule-40
If there is continuous abstraction in an open unconfined aquifer
then there will be 100% of the groundwater abstraction at the expense of the river flow.

Rule-44
If a borehole is put down close to the springhead

APPENDIX C: EXAMPLE OF RULE IMPLEMENTATION IN ELSIE

```
/*
*                       ProTalk Rules for Task Source Of Supply
*/

#include <prk/lib.pth>

/*******************************/
/*ruleset SourceOfSupplyRules      */
/*******************************/

bcrule CheckSourceOfSupply in SourceOfSupplyRules priority 300
{
if:
        ?sourceconfirmed = technicalassessmentresults.ConfirmedSourceOfSupply;
        ?sourceconfirmed != Null;
        ?supportedsource = technicalassessmentresults.CheckedSupportedSources;
        ?supportedsource != Null;
then:
        technicalassessmentresults.CheckedSourceOfSupply = Yes;
        Warning("Checked all source of supply details!",
            "Any problems reported must be handled immediately", Wait);
}

bcrule ConfirmSources in SourceOfSupplyRules priority 200
{
if:
        technicalassessmentresults.ConfirmedSurfaceWaterSource == Yes;
        technicalassessmentresults.ConfirmedGroundWaterSource == Yes;
then:
        technicalassessmentresults.ConfirmedSourceOfSupply = Yes;
        Warning("Source of supply confirmed!","", Wait);
}

bcrule NoConfirmSources in SourceOfSupplyRules priority 10
{
if:
        technicalassessmentresults.ConfirmedSurfaceWaterSource != Yes;
        technicalassessmentresults.ConfirmedGroundWaterSource != Yes;
then:
        technicalassessmentresults.ConfirmedSourceOfSupply = No;
        Warning("Source of supply details wrong!",
"The source details are not correct. Suggest review of
source points in application form.", Wait);
}
```

```
bcrule NoConfirmGWSource in SourceOfSupplyRules priority 1
{
if:
            technicalassessmentresults.ConfirmedGroundWaterSource != Yes;
then:
            technicalassessmentresults.ConfirmedGroundWaterSource = No;
            Warning("Groundwater source of supply not confirmed!",
                        "Error in application form details?", Wait);
}

bcrule NotSupportedSource in SourceOfSupplyRules priority 150
{
if:
            ?source == direct instanceof SurfaceWaterSourcePoints;
            ?source.Type != River;
            ?source.Type != Tributary;
            ?source.Type != Canal;
            ?source.Type != Aqueduct;
then:
            ?source.SupportedRiver = No;
            ?legend = ?source.Legend;
            ?ngr = ?source.NGR;
            ?message = AppendStrings("Source point ", ?legend, " at NGR ", ?ngr,
                                                " is not a supported source of supply");
            Warning("Unsupported source of supply!", ?message, Wait);
}

bcrule SupportedRiver in SourceOfSupplyRules priority 100
{
if:
            ?source == direct instanceof SurfaceWaterSourcePoints;
            ?supportedrivers = all RegionalData.SupportedRivers;
            ?name = ?source.SourceName;
            ?item inlist ?supportedrivers;
            `(?name, ?upstream, ?downstream) == ?item;
            ?namestring = ConvertToString(?name);
            ?question = AppendStrings(?namestring,
" is a supported source of supply in places.
Is the abstraction point between ", ?upstream, " (upstream limit)
and ", ?downstream, " (downstream limit)?");
            ?yn = InCodeAskForOneOf(?question, `(Yes,No), Single, No,
                "Attempting to determine if we have a supported source of supply");
            ?yn == Yes;
then:
            ?source.SupportedRiver = Yes;
            ?legend = ?source.Legend;
            ?ngr = ?source.NGR;
            ?message = AppendStrings("Source point ", ?legend, " at NGR ", ?ngr,
                                                " is a supported source of supply");
            Warning("Supported source of supply!", ?message, Wait);
}
```

APPENDIX D: ATTENDEES OF THE ELSIE USER TRIALS

Gary Arkell. Licensing Assistant, Thames Region
Andy Barron, Water Resources Officer, Thames Region
Mervyn Bramley, NRA Head Office
Roz Barraball. Licensing Assistant, Thames Region
Geoff Bell, Abstraction Control Manager, Thames Region
Sarah Douglas, Hydrologist, Sussex Area Office
John Ellis, Regional Resources Management Officer, Southern Region
Sheila Greenfield, South Western Region
Gordon Hargreaves. Anglian Region
Peter Herbertson. Regional Water Resources Manager, Southern Region
Keith Hunter, WTi
Debbie Jones, Anglian Region
Mike Owen, Water Resources Business Manager, Thames Region
Wendy Rogers, Licensing Officer, Sussex Area Office
Graham Tanner, Senior Licensing Assistant, Thames Region
Richard Westaway, South Western Region
Stephen White. Resources Technician, Southern Region