THE APPLICATION OF EXPERT SYSTEMS IN GEOGRAPHICAL INFORMATION SYSTEMS

H. M. UHLENBRUCK

June 1988

TECHNICAL REPORT
NO. 138

PREFACE

In order to make our extensive series of technical reports more readily available, we have scanned the old master copies and produced electronic versions in Portable Document Format. The quality of the images varies depending on the quality of the originals. The images have not been converted to searchable text.

THE APPLICATION OF EXPERT SYSTEMS IN GEOGRAPHICAL INFORMATION SYSTEMS

H. Matthias Uhlenbruck

Department of Surveying Engineering
University of New Brunswick
P.O. Box 4400
Fredericton, N.B.
Canada
E3B 5A3

June 1988

Latest Reprinting May 1993

PREFACE

This technical report is a reproduction of a thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Engineering in the Department of Surveying Engineering, March 1988.

ABSTRACT

Geographical information systems (GIS) are excellent tools for an inventory of land-related data. GIS users must have an in-depth knowledge about the system in order to make effective use of the system to analyze these data. Making these systems more intelligent and easier to use is therefore an ongoing primary research topic.

Expert systems are considered a solution to making GIS more powerful and user friendly. They can be applied to emulate reasoning processes requiring expert knowledge on a computer. This study evaluates the applicability of expert system technology to support data analysis with a GIS. The ranking of groundwater wells by their needs for protection against contamination is used as an example to demonstrate the integration of GIS and expert systems. The knowledge for a complex data analysis is contained in the expert system while the GIS provides the functions of managing and analyzing spatial data.

A prototype for the above mentioned system was developed during the study. This prototype has been implemented using an expert system building tool (expert system shell) and a commercially available GIS. A groundwater supply inventory, carried out for the Province of New Brunswick, provided the basis for the development of the automated well ranking system.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AI	=	Artificial Intelligence
CARIS	=	Computer Aided Resource Information System
CARED	=	CARis EDitor
CARMAN	=	CARis MANager
DBMS	=	Data Base Management System
EQUEL	=	Embedded QUEry Language
ESQL	=	Embedded Structured Query Language
GIS	=	Geographical Information System
INGRES	=	INteractive Graphics and REtrieval System
KB	=	Knowledge Base
LHS	=	Left Hand Side (of a rule)
QUEL	=	QUEry Language
RHS	=	Right Hand Side (of a rule)
SQL	=	Structured Query Language
WRPB	=	Water Resources Planning Branch

Acknowledgements

A number of people have contributed to the successful completion of this project:

For his guidance in research and preparation of this thesis I want to thank my supervisor Prof. John McLaughlin. Helpful suggestions about geographical information processing were provided by Prof. Y.C. Lee and about expert system technology Prof. Brad Nickerson. I would also like to thank Prof. Dale Bray, Lawrence Peters and Alan Gregory for supplying information and ideas on groundwater problems. Acknowledgements are also made to Prof. Sam Masry from Universal Systems Ltd. for the provision of computer equipment and to his staff for their helpful assistance.

I would also like to express my gratitude to all those who helped in the proofreading of this manuscript and to those who gave me suggestions on style and presentation of this thesis. My final thanks go to Prof. Rodney Cooper who gave me the initial idea for this project.

CHAPTER 1

INTRODUCTION

1.1. Geographical Information and Expert Systems

Geographical information systems (GIS) have been under development for more than 20 years. They are designed to cope with graphical and textual information about an area within an integrated system. Some systems place the main emphasis in their performance on the cartographical quality of their products, others on the analysis of geographical data. They are generally accepted within the community of users of geographical information as excellent tools to process large amounts of heterogeneous spatial data.

The analysis of spatial data for decision support is a complex process, which requires a considerable amount of knowledge and experience. With the GIS, a tool is available which gives the user the analytical power of a computer for overlaying, displaying, integrating and storing data collected from a variety of sources. This aid, however, requires not only competence in the area of analysis, but also the ability to make effective and efficient use of the GIS. This fact poses restrictions on the general applicability of GIS technology. Significant research efforts are undertaken by the people involved with spatial information systems to overcome this problem.

The application of expert system technologies as a way to solve the above mentioned problem has been under discussion for the last few years. Expert systems are programs which use a computer model of human reasoning techniques. They apply this model to expert knowledge stored in the computer, to come up with the same conclusion a human expert would reach, if faced with the same problem [Weiss and Kulikowski, 1984]. For GIS applications, an expert system could support the data analysis in two ways:

1) by providing a more intelligent user interface and thus relieving the expert from the necessity of having in-depth knowledge about the system; and
2) by capturing the expert's knowledge in a form that is storable and transportable, and thus providing the possibility of data analysis without the necessity of the expert carrying out the analysis by himself in all cases.

The incorporation of expert system technology into GIS has been attempted recently in several projects. One of these attempts, ASPENEX, actually became implemented to support the forest management of aspen stands in the Nicolet National Forest in Northern Wisconsin [Morse, 1987]. Many researchers and users feel that the future research needs of geographic and/or land information systems will likely be resolved within the context of expert systems [Robinson and Frank, 1987]. The publication by Robinson and Frank gives a good overview of different aspects of expert system/GIS research and of projects that have been and are under investigation.

1.2. Expert System for Well Site Ranking

Geographical information systems are an important area of research at the Department of Surveying Engineering, University of New Brunswick. Within the Resource Information Management Group, the interest is mainly focused on the application of GIS. One of these applications is the ranking of groundwater wells into categories of pollution protection needs. This problem was chosen to be implemented using an integrated expert system/GIS. The ranking of the wells requires a significant amount of expertise, which is to be captured with the expert system. Most of the information needed for the ranking is of a spatial nature and can thus be stored, analyzed and retrieved with a GIS.

Groundwater contamination is an area of concern for the Water Resource Planning Branch (WRPB), New Brunswick

Department of Municipal Affairs and Environment. Approximately 70% of New Brunswick's population uses groundwater as its freshwater source. Several cases of contamination have caused WRPB to undertake a study, which establishes an "Inventory and Review of New Brunswick Municipal Groundwater Supply Areas" [Gregory, 1988]. At the same time, WRPB carried out a "Groundwater Information Pilot Project" with the objective to demonstrate the effectiveness of a GIS for groundwater applications [WRPB, 1987].

The pilot project mainly provided the graphical part of the database used for the expert system/GIS implementation, while the textual data were provided by the inventory. The necessary knowledge for the well site ranking could be extracted from preliminary reports on the review and from discussions with groundwater experts.

1.3. Objectives and Structure of Study

The main objective of this study was to show the possibility of enhancing GIS capabilities by using expert system technology. Building an expert system from scratch is a difficult task and would be a research topic by itself. Expert system building tools could, however, simplify this task significantly, but leave the system designer less freedom. A second objective, therefore, became the testing of the applicability of a generic tool to build the system and link it to a GIS. The third objective, finally, was to demonstrate the capability of an integrated expert system/GIS to rank well sites by their protection needs in a way that simulates the process carried out by a human expert.

Preliminary research was carried out on the theory of expert systems, which is documented in Chapter 2 of this thesis. Based on this research, an expert system building tool was selected, which is briefly described at the end of the chapter.

Chapter 3 gives an extract of the review as it was carried out for WRPB [Gregory, 1988]. The chapter mainly focuses on the elements of this review that were used for the implementation of the inventory on the GIS and the well ranking procedure of the expert system. It also describes refinements, which had to be made to the expert's well ranking scheme in order to simulate the process on an expert system.

Chapter 4 describes the system itself, as it has been implemented during the study. After describing the implementation environment, the main emphasis is placed on the presentation of the interaction between expert system and GIS. A simple way of combining the two systems is introduced.

The concluding Chapter 5 summarizes the research work done for the project. It describes the advantages and disadvantages of the method chosen for the implementation. The report ends with recommendations for future work on expert system/GIS integration and other expert system enhancements for the management of land information in a geographical information system.

CHAPTER 2

ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

2.1. Introduction

Intelligent machines appear frequently in mythology and science fiction. To invent a machine that simulates or even excels human capabilities always has been a dream of many people and this concept is expressed in numerous articles, books and films [McCarthy, 1983].

Systematic work on the simulation of intelligence began only after the invention of digital computers. With the computer, a machine was developed which can perform calculations and logical processes that would take years if done by humans, but which can be performed in seconds or minutes by these machines. A tool was invented that can be considered as "extending the intellect of human beings as the bulldozer extends their muscles" [Bashkow, 1983]. From the beginning, scientists tried to simulate human thinking processes using the computer. The study of "how to make computers do things, which, at the moment, humans are better at" is called Artificial Intelligence (AI) [Rich, 1983].

The British mathematician Alan Turing wrote the first serious scientific article about AI in 1950 [McCarthy, 1983]. Stanford University stands out as the institution that triggered AI development with its Heuristic Programming Project in the 1950's [Kumara et.al., 1986]. Today, AI is a factor in many research institutions and is applied in various fields, but it is still in its infant stage. The U.S. artificial intelligence market last year amounted to $256 million or 0.2% of the total computer market. It is forecast that this market will grow rapidly, reaching $113 billion by the year 2000, one quarter of the total computer market [Wiig, 1984; Pepper, 1987].

Intelligence can be defined as "the ability to learn or understand from experience (and) to respond successfully to new situations" [Coles, 1979]. The human brain is equipped with an enormous potential to perceive, understand and learn. One can call a computer intelligent, if it can duplicate this ability. The degree to which this duplication succeeds gives a degree of intelligence of a computer system.

Turing proposed a simple test to determine whether a computer succeeds in duplicating the human reasoning process: A person communicates simultaneously with a computer and another person. If at one point in the dialogue he cannot differentiate between his human partner and the computer, it can be said that the computer has performed at the level of a human being.

2.2. Principles and Applications of AI

Chess playing was one of the first AI applications that were successfully implemented. It displays one of the core principles of AI, the concept of intelligent search. For each move in a chess game there are many choices. This leads to a combinatorial explosion of possibilities through an average game of 40 moves (approximately 10^{120} possibilities) [Kumara et.al., 1986]. Testing all possibilities cannot be achieved in a reasonable response time even with today's supercomputers. The search must be restricted. Heuristic rules are employed to reject most alternatives. One such heuristic, called "alpha-beta" [McCarthy, 1983], stops examining possible consequences of a move as soon as it finds one reply refuting it. Incidentally, the same method is also used by human players, albeit often subconsciously.

The main issues in AI research include a) what information a program should have and how to store it in the computer, and b) how further conclusions can be drawn from initial information. Mathematical logic provides powerful methods for both the representation of knowledge and ways of reasoning. Logic studies the relationship of implication between assumptions and conclusions [Kowalski, 1979]; it is a systematic way of reasoning [Gray, 1984]. Additionally, mathematical models representing "real-world reasoning" (e.g. Zadeh [1979], with "fuzzy sets/fuzzy logic") make it possible to simulate the way humans often reach conclusions when confronted with incomplete or imprecise initial information.

The field of artificial intelligence can be subdivided into a number of sub-areas. The following paragraphs describe some of them briefly (for a more thorough review see e.g. Jackson [1974], Feigenbaum et.al. [1982] or Bernhold and Abbers [1984]):

- Pattern Matching

  Definition: Programs that recognize objects by comparison with stored patterns in a database.
  The system should be able to identify identical and similar objects. Changes in lines, colors or brightness are used to analyse features of objects and correlate the resulting feature vector to the feature space stored in the database. In the field of surveying engineering, pattern recognition is of interest to identify objects in remotely sensed scanned data or scanned maps.

- Natural Language Processing

  Definition: Program systems using a grammar syntax and a dictionary of words together with a semantics interpreter.

  The validity of sentences is checked using the grammar rules. The semantics interpreter analyses the meaning of a sentence. Practical applications include comprehending text, translating into another language or answering queries from databases posed in natural languages. A closely related field is speech recognition and speech understanding.

- Robotics

  Definition: Machines that are employed to carry out strenuous or dangerous tasks, which cannot be done or are not desired to be done by humans.

  Applications include agricultural harvesting, factory material handling and transfer, combat and combat support systems and planetary exploration vehicles. The robots must be able to perceive and adapt to their environment (e.g. using video image processing techniques) and report successes and failures [McTamaney, 1987]. Today's robots deal with very specialized tasks. In the future, easily programmable, general purpose robots are planned to be used for manifold applications.

- Expert Systems

  Definition: Computer programs applied to emulate reasoning processes requiring expert knowledge and experience.

  The systems consist basically of a database of data and knowledge, and a system that controls the application of this knowledge to analyze the data. Expert systems have been successfully employed in areas of expertise such as medical diagnosis, mineral exploration and computer configuration. Interaction between expert systems and existing, large databases is a very active area of research in the computer science community [Kerschberg, 1984].

Interaction or floating boundaries between the above mentioned and other AI sub-areas are common and will become even more important in the future. In order to simulate human performance, an automated system must be able to act intelligently in more than only one narrowly defined area. Almost all expert systems, for example, have some kind of natural language processing capabilities for their user interface.

For a system to be considered truly intelligent, it must be able to learn during its use. It is often claimed that a system is not intelligent until it can learn by experience [Schank, 1983; Michalski et.al., 1983]. Very few computers, however, presently exhibit learning capabilities [Nickerson, 1988]. The field of machine learning is therefore becoming increasingly important in all sub-areas of artificial intelligence.

2.3. Expert System Technology

The most successful application of AI is <u>knowledge-based</u> or <u>expert systems</u>. An expert is a person who has aquired extensive knowledge in a certain area by way of education and/or experience. Knowledge and experience enable the expert to solve specific types of problems. Decisions are made by considering the information given on a specific problem together with existing data on the problem area. Expertise then provides the means to analyse the problem and arrive at a solution. In many cases other experts must be consulted to deal with complex problems. Each of the decision making processes adds to the knowledge of the expert and can be used in future decision making [Uhlenbruck and McLaughlin, 1987].

Expert systems recreate the human reasoning process and emulate the decision making process of a human expert. They:

1) handle real-world, complex problems requiring an expert's interpretation, and

2) solve the problems using a computer model of expert reasoning, reaching the same conclusion that the human expert would reach if faced with the same problem. [Weiss and Kulikowski, 1984]

Figure 2.1 shows the different modules of which an expert systems is comprised. These modules represent counterparts to essential factors of human expert reasoning: The knowledge_base corresponds to knowledge and experience of an expert, whereas the global_database contains existing data and hypotheses on the problem area. The inference___engine accommodates reasoning methods simulating the way human experts would apply their knowledge to analyse information and reach a decision. The user_interface provides for communication between the user and the system.

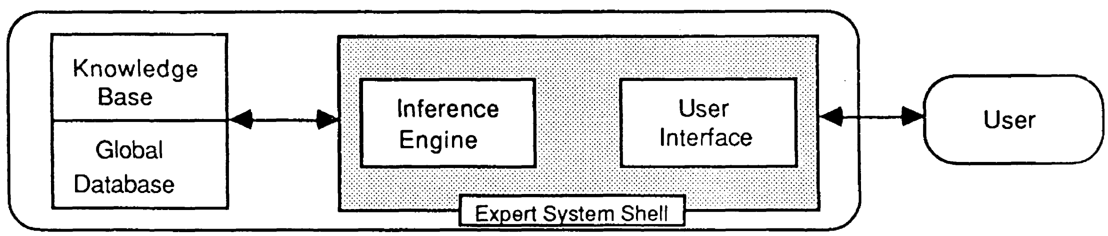

Figure 2.1: Expert System Modules

Today's expert systems normally have the following three characteristics [Nickerson, 1987a]:

1) They deal with a specific, focussed task having a relatively narrow range of applicability.

2) Knowledge is kept separate from reasoning methods used to draw conclusions.

3) They are able to explain their actions and lines of reasoning.

It is the separation between knowledge base and inference engine which makes expert systems much more versatile than other computer programs. Knowledge and control of the program are built and maintained individually. The knowledge base can be modified independendly from the inference engine. The following paragraphs describe the different expert system modules and their interaction in more detail.

2.3.1. Knowledge Base and Knowledge Acquisition

Most expert systems store their knowledge in the form of rules. They are therefore also often called "*rule-based*" systems. The most common form is the storage of IF .... THEN .... type rules. The left-hand-side (LHS) of these rules is comprised of one or more conditions or antecedents. On the right-hand-side (RHS) are one or more propositions or consequents. The expert knowledge is formalized and structured into these rules.

Another form of knowledge representation are frames. Information is grouped in terms of records of slots and fillers [Nickerson, 1987a]. (For a more detailed description see Barstow et.al. [1983]). The process of extracting knowledge from an expert and structuring it into rules or frames is called "*knowledge acquisition.*"

The type of knowledge used by experts to solve problems is often subjective, ill-codified and partly judgmental

[Buchanan et.al., 1983]. In most cases, it is not formulated in a fashion that is easily translatable into a program. The difficult task of extracting the expert's understanding of a problem and representing it as facts and relations in an expert system is often carried out by a "knowledge engineer."

The approach adopted to structure knowledge in an expert system depends on the application area. The knowledge representation can vary drastically from system to system. One type of problem often solved with the help of expert systems is a classification problem. The related knowledge representation is called a classification model. In such a model a conclusion is selected from a pre-specified list of possibilities. In the abstract this implies three separate lists contained in the knowledge base [Weiss and Kulikowski, 1984]:

1) a list of possible conclusions (Hypotheses)
2) a list of possible observations (Data)
3) a list of rules relating observations to conclusions

The process of building an expert system forces the human expert to go through the decision making process in an ordered and logical fashion. This can be very helpful in discovering aspects of the problem that have not been considered before. Thus, the building of an expert system is not only a project to collect expert knowledge for decision support in an automated system, but also helps to refine this knowledge and structure it in a logical manner.

Figure 2.2 shows a schematic process of acquiring knowledge for an expert system and the people involved. The maintenance of a knowledge base practically never stops throughout the lifetime of a system, much like a human expert refines his knowledge and methods of reasoning over the length of his professional activity.

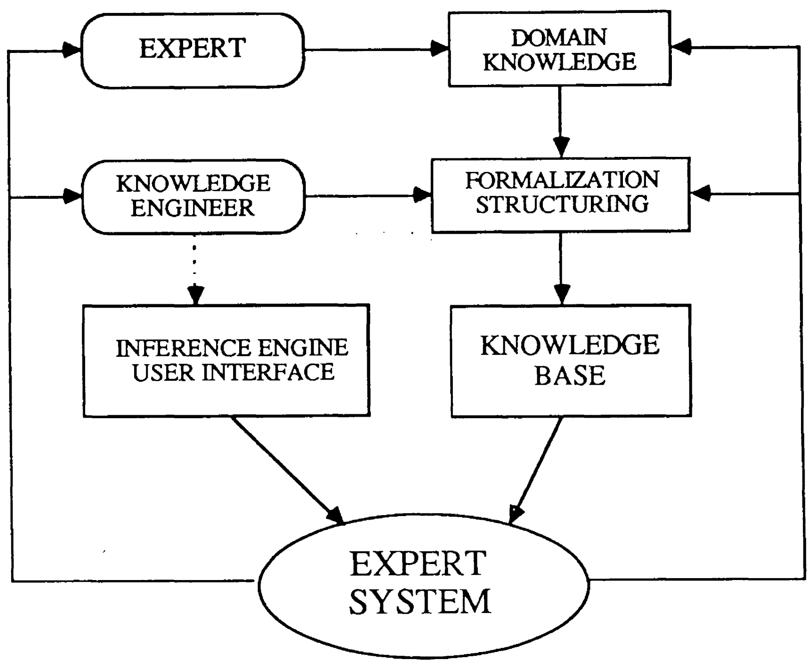

Figure 2.2: Process of Knowledge Acquisition

Knowledge contained in an expert system can be categorized into two classes: a) the domain knowledge and b) the general problem solving knowledge. While the domain knowledge is contained in the knowledge base, the methodology of how to solve problems in general, which is applicable to different domains of expertise, is coded in another module of an expert system, the inference engine.

Yet another type of knowledge can be identified: knowledge about the knowledge in the system, which is called the metaknowledge [Lenat et al., 1983]. Two examples of meta-knowledge are:

M-Rule 1: Prefer experts' rules to novices' rules.

M-Rule 2: IF aquifer type is "confined",
THEN don't try rules containing "water table".

Metaknowledge can be used to manipulate the firing (that is the execution of RHS actions after all LHS conditions are proven to be true) of one rule over another, or to direct the path of a system through the application of the rules. It is employed to run an expert program more efficiently and lead to the most accurate solution.

In addition to knowledge contained in rules, which comprises the reasoning knowledge of a system, some systems allow for the representation of knowledge about the data items or objects the rules deal with. In fact, frame-based systems are built mainly on the description of these objects.

Each data item occurring on the LHS or RHS of a rule can be described in detail in the object representation part of the knowledge base. Relationships between objects are defined here and a network of objects and subobjects with their respective properties can be built. The reasoning can thus affect whole classes of objects, adding a new dimension to the knowledge processing and making it more powerful. This method of reasoning is also often referred to as deep reasoning.

2.3.2. Dealing with Uncertainties

Information used to derive a decision which solves a problem is often afflicted with uncertainty. In addition, an expert often has to solve a problem without having a sufficient amount of data, using conflicting information or unreliable knowledge for interpreting the data [Stefik et.al., 1983]. In order to simulate the human decision making process, an expert program must be able to cope with these uncertainties on its path of reasoning.

Statistical methods, based on probability theory, do not appear to be a solution to the problem, due to complex interdependencies of propositions and conditions in different rules [Weiss and Kulikowski, 1984]. Most systems, therefore, represent uncertainties in a "pseudo-probabilistic" form called confidence factors. These consist mostly of numbers greater than zero for positive evidence and numbers less than zero for negative evidence. Rules and facts in the database have some confidence factors associated with them. The combination of these leads to a conclusion with a specific degree of certainty.

Another approach to handling uncertainties is based on fuzzy logic [Turner, 1984]. This method is claimed to provide more general means of approximate reasoning. An overview of approximate reasoning techniques, which have been proposed for dealing with uncertain or imprecise knowledge in expert systems, can be found in Prade's [1983] paper.

Additional rules, expressing the lack of essential information, can further weigh down the degree of certainty established for a solution determined by the system. Confidence factors for rules and facts together with these additional rules account for the capability of an expert system to deal with uncertainty.

2.3.3 Global Database and Inference Engine

The global database provides a working storage during the evaluation of rules. At the beginning of an expert system session, it usually contains a hypothesis to be proven and a few data known initially about the problem.

The information is constantly updated and stored here temporarily until the end of the expert system run. In programs that combine knowledge from different expert systems, the global database is often referred to as the _blackboard._ The information gained or derived by the different programs is stored and retrieved from here, with the blackboard acting as the link of data flow between several systems.

The inference engine is the heart of an expert system. It controls the execution (firing) of rules leading to a conclusion. In this program module the contents of the global database are matched against the contents of the rule base. Rules matching the elements of the database are fired. During an expert system run more than one rule at a time might be applicable. In this case a conflict resolution strategy must be used to determine the rule to be fired first. These strategies can include the following:

1) a rule is not allowed to fire more than once on the same data (refractoriness);
2) rules using more recent data are preferred to rules which match against data that have been in the global database for a longer time (recency); or

3) rules with a greater number of antecedents are preferred to more general rules (specificity) [Nickerson, 1987a].

Two basic control mechanisms are normally used:

1) Forward chaining
2) Backward chaining

A combination of both is also sometimes used.

Forward chaining systems progress from the given information (data or event) to a conclusion. They are therefore sometimes called "event driven." The facts about a problem, provided either directly by an interactive user, from an external database or program, or from other rules, are stored in the global database comprising the current state of knowledge about a problem [Brownston et.al., 1985]. The inference engine determines the rules containing the data in their LHS and evaluates these antecedents by matching them against the contents of the database. If this evaluation determines all conditions to be true, then the consequent actions (RHS) are executed leading to an update of the global database. The system proceeds to invoke the rules in a forward direction, continuing until the problem is solved, or no further rule can be invoked [Weiss and Kulikowski, 1984]. The latter means that the problem is unsolvable with the existing information.

Backward chaining starts with the conclusion (hypothesis or goal) of the problem, searching the consequents of all rules for occurrences of the goal. Systems employing this technique are therefore sometimes called "goal driven." If at least one rule is found with the desired goal, then the antecedents of this rule become the hypotheses and are recursively evaluated [Nickerson, 1987a]. The process stops when the problem is solved or it is proven that it cannot be solved after firing all possible rules.

2.3.4. User Interface

In most cases a great amount of effort is spent to provide a user-friendly interface. The capability of the system to process natural language plays an important role. Many systems interact with users by way of normal English sentences for questions and answers.

In order to make the system acceptable as an expert, it must be transparent, that is, it must be able to answer a) why a particular question is asked; and b) how a certain conclusion has been derived. Users should have the feeling that they are "talking" to a real expert, who is able to explain his behaviour. The why/how utility is also a good tool in the developing stage of an expert system when debugging the knowledge base.

2.3.5. Programming Languages and Expert System Shells

In the beginning years of expert system research, the need for a non-procedural programming language was expressed. Many AI applications use a list processing language, LISP, developed at the Massachusetts Institute of Technology [Winston and Horn, 1981]. LISP, however, is still a language in which the programmer expresses how to do things. In this concern, it is like conventional computer languages, such as FORTRAN, even though it is much more expressive. It is the "champion" of these convetional languages, as Bratko [1986] puts it.

PROLOG (PROgramming in LOGic), a language developed in the early 1970's based on the theories of Kowalski [1979], is, as the name indicates, based on logic. As stated in Section 2.1, mathematical logic is the fundamental basis upon which artificial intelligence and expert systems are built. PROLOG, therefore, lends itself as a natural language for building expert systems, with its structuring into clauses of facts, rules and questions [Bratko, 1986]. Most PROLOG programs, however, are interpreted instead of compiled and run relatively slowly.

The majority of today's AI applications are written in LISP. They often run on special "LISP machines" and are therefore not easily transportable. Recently, more and more systems are developed in the "C" programming language. C is a basic language, which is considered "closer to the machine" and therefore faster than LISP or PROLOG. Some programs, originally written in LISP or PROLOG, are today being converted into C.

The inference engine and the user interface are often seen as one unit, a program called the expert system shell. The shell can be developed independently from the knowledge base. It represents a generic expert system containing in its knowledge base only the general problem solving knowledge as described in Section 2.3.1. Upon this, one can build expert systems for different knowledge domains, using a predefined type of knowledge representation. Many different shells can be purchased today, significantly simplifying the process of building an expert system.

The shells usually feature a rule-editor for building and debugging the knowledge base and an inference engine for forward/backward chaining or combinations of these. Almost all tools include a why/how utility. More sophisticated shells feature different schemes of knowledge representation (e.g. object description, combination of frames and rules), graphic interfaces (e.g. to show the network of rules), and various degrees of natural language processing capabilities. This is very helpful in developing the system and understanding its reasoning processes. A very important feature of expert system shells is the interfacing capability between the system and external databases or programs. Good shells often have dedicated interfaces for full integration of database management systems and links to several programming languages.

The shells are available for all types of computers from personal computers (PC) to mini computers and mainframes. Prices range from as low as $400 (U.S.) for the PC up to $100,000 or more for mainframe computers.

2.3.6. Introduction to an Expert System Shell

The development of an expert system for this project was realized using an expert system shell. Since the communication to other programs, residing on a MicroVAX[1] II under VMS[1], was a major concern of this study, a shell was desired which could run under VMS. After several weeks of researching the market, NEXPERT/OBJECT of Neuron Data Inc. was selected. NEXPERT features most characteristics of top-of-the-line shells in spite of being significantly lower in price. Versions are available for several VAX minicomputer models, as well as

---

[1]MicroVAX and VMS are trademarks of Digital Equipment Corporation.

the IBM AT or compatible PCs, and the Apple MacIntosh. The heart of the system is comprised of a number of subprograms (sharable images), which are called from a graphic interface and external routines.

The shell is a hybrid system allowing for the representation of knowledge through both rules and objects. The following brief description[2] of some characteristics of the shell are based on the user's manual "NEXPERT/OBJECT Fundamentals" [Neuron Data, 1987].

The LHS of a rule consists of one or more conditions. These conditions have the form of 3-tuples comprised of an operator, an attribute and a value. Attributes can have either character, numeric or boolean values. Operators relate the attributes to their values. This relation establishes the premise of a condition, which has to be checked during the system run to see whether the condition is true or false.

The RHS contains the hypothesis of a rule. This hypothesis has a boolean value (true or false). If all LHS conditions are found to be true, then the hypothesis is true. In this case, zero or more actions can be

---

[2]The product description given here reflects only features which were used during this project. Many capabilities of the system, especially the complexity of the knowledge base, could be explored only on the surface. A more scrutinizing examination would be the topic of a separate study.

performed. These RHS actions have the form of 3-tuples similar to the conditions. They can include, among others, the loading of new rules, the execution of an external program, or the display of graphics or text.

Table 2.1: NEXPERT Rule

```
IF (LHS conditions)
      IS    aquifer.susceptibility   high
  and IS    well_replacement.cost    moderate
  and YES   landuse.hazardous

THEN (RHS hypothesis)
            pollution_threat
AND (RHS actions)
      DO        well.group   3
  and EXECUTE   showmap      @ATOMID=well.name
```

__Interpretation:__
The LHS of the rule is concerned with the objects aquifer, well_replacement and landuse, and their respective properties susceptibility, cost and hazardous, separated from their objects by decimal points. Susceptibility and cost have string values, while hazardous has a boolean value. If the susceptibility of the aquifer is high, the cost of replacing the well moderate, and there is hazardous landuse, then the rule fires. Pollution threat is set to true and the RHS actions are executed, i.e. the well is ranked into group 3, and the external routine "ShowMap" is called with the NEXPERT ID of the name of the well as the only argument.

All attributes of the rules are considered __data.__ These data are properties of objects or classes of objects. They are described in the second part of the knowledge base, the __object representation.__

For each property there exists a _meta-slot._ These meta-slots specify how a data item can obtain its value, and also allow for the definition of several actions upon modification of this value during the expert system run. One can specify the importance of a property and thus the point in time during knowledge processing when the value should be obtained relative to other properties. An _order of sources,_ giving several choices of where to find the value, can also be specified.

Every entity (e.g. objects, rules, etc.) in NEXPERT is considered an _atom_ (see Figure 2.3). Each atom has an _identification number serving_ as a logical descriptor used internally by NEXPERT. External routines communicating to NEXPERT use these IDs to refer to a certain atom.

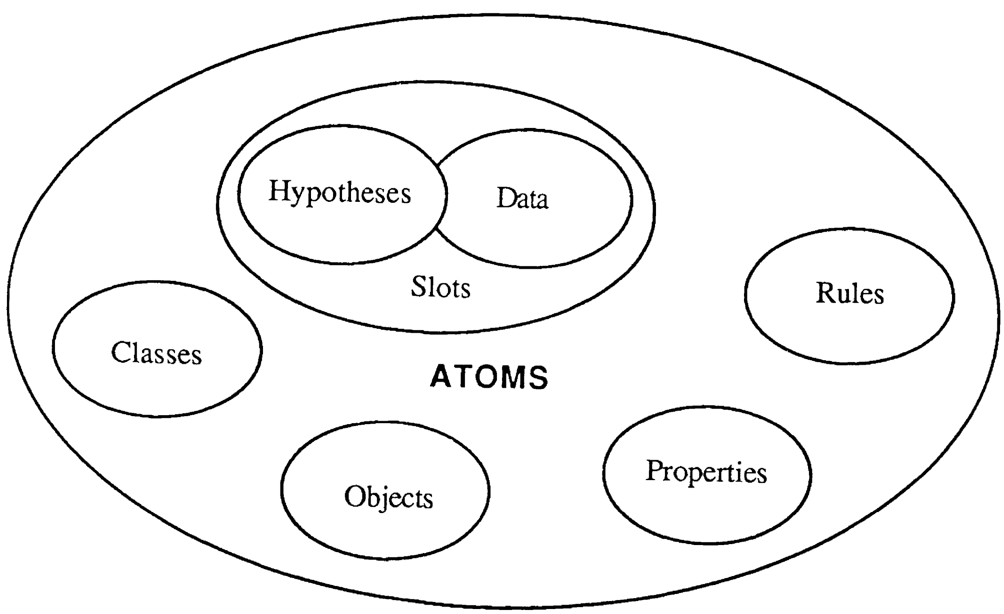

Figure 2.3: NEXPERT Atoms

NEXPERT operates with a graphic interface for both creation of a knowledge base and knowledge processing. This interface makes the system very easy and illustrative to use, but it also poses some restrictions on the hardware environment. The VAX version, for instance, runs only on a VAXStation[1] or a GPX[1] terminal.

Both parts of the knowledge base, rule and object representation, can be made visible using the graphic interface. Rule and object networks can thus be investigated and the interdependencies between them shown graphically. Rule, object and meta-slot editors can be called from the graphic networks, which is of great help during the system development phase, and also to better understand reasoning processes.

Initially, all data and hypotheses in the system are in a state of UNKNOWN. To run the expert program one can either <u>volunteer</u> some data to initiate the system for forward chaining, or <u>suggest</u> a hypothesis to initiate the system for backward chaining. The system then follows a combined forward/backward chaining path to solve the problem it was originated to work on. On this path, values for conditions and hypotheses are established.

----

[1]VAXStation and GPX are trademarks of Digital Equipment Corporation.

Each condition is checked whether its premise is matched by the value found for its attribute. Attribute values can be obtained from the sources given in the meta-slots, or, as the default, by asking the user of the system in an interactive dialogue. If premise and attribute value match, the condition is true, otherwise it is false. A condition can also get a value of NOTKNOWN, if it is tested, but its attribute value is not known. If a condition is not checked at all, its value remains in a state of UNKNOWN.

The values of the conditions reflect on the values for their hypotheses. If all conditions of a rule are true, the hypothesis is true and the actions are executed. If only one condition is false, the hypothesis is false. If there is no false condition, but one or more with a value of NOTKNOWN, then the hypothesis value is also set to NOTKNOWN. A hypothesis remains UNKNOWN if no rule containing it is considered during the reasoning process.

At the end of a system run, users have several choices to display the results. They can:

a) retrieve a summary of values for data and hypotheses that were processed,

b) retrieve a full report on the knowledge processing, showing rules that were fired and those that were rejected, or

c) display the network of rules showing depictive icons which indicate whether conditions, hypotheses and actions were determined to be true, false, unknown or not known.

CHAPTER 3

GROUNDWATER CONTAMINATION AND WELL SITE RANKING

3.1. Groundwater Utilization and Need for Protection

Together with land and air, water is one of the vital components for all forms of life on earth. Plants and animals, including human beings, require water to carry out their life functions and in fact consist mainly of water. The human adult, for example, consists 65%-70% of water [Davenport, 1983]. From this it is obvious, that water as a natural resource is of extreme importance for mankind.

Approximately 97% of all water resources is salt water, contained in the oceans and unusable for human consumption, unless subjected to a costly treatment (see Figure 3.1). From the remaining freshwater, two thirds are contained in ice caps and glaciers, leaving only one percent of the total water resources to be utilized. Only a small amount of this water (approx. 2%) is available on the surface of the earth as lakes and rivers. Ninety-eight percent of earth's fresh liquid water is contained under the surface [Price, 1985]. This groundwater, therefore, represents an extremely important resource.

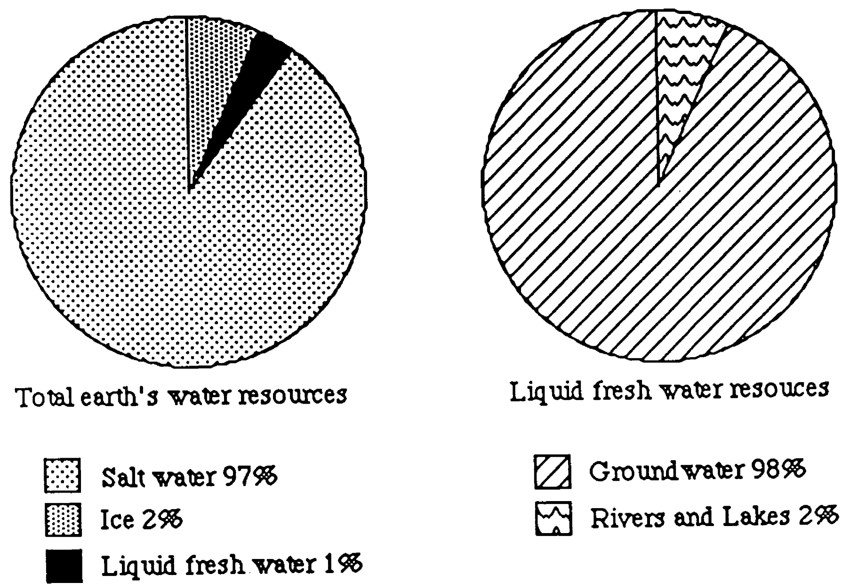

Figure 3.1: Distribution of Water Resources on Earth

Even in Canada, a country with enormous amounts of surface water resources available in the form of lakes, the use of groundwater is becoming more and more popular. As Environment Canada statistics show, more than 6.2 million or 26% of Canadians relied on groundwater for their domestic supply in 1981, up from 10% in the 1960's [Hess, 1981]. In the province of New Brunswick, about 70% of the population rely on groundwater as their source of fresh water [Peters, 1988].

Groundwater has substantial economic advantages over surface water for drinking purposes. It can often be developed where and when it is needed by sinking boreholes in appropriate places. A surface reservoir, on the other hand, must be developed at times when its full capacities are not needed. It occupies a large area and often conflicts with other land uses. Additionally, groundwater is protected from evaporation which can cause substantial losses of water from reservoirs and lakes [Price, 1985]. A further advantage of groundwater over surface water is the higher vulnerability of the latter to pollution. Several widely publicized spills, e.g. the discharge of water containing toxic chemicals after a fire at Sandoz, Switzerland in 1987, exemplify this fact.

Buffered by soil and rock, groundwater is less vulnerable to pollution than surface water. Nevertheless, groundwater pollution occurs and this pollution can have a significant impact on water supplies. Once contaminated, the containment of the water in the ground reveals another problem: a clean up is extremely difficult. Due to the relatively slow flowrates of groundwater, wells will remain unusable for an extended period of time after the pollution has affected the aquifer (periods of over 100 years are possible).

Prevalent problems in the Maritime Provinces have been gasoline and oil spills: in 1979 there were 19 spills recognized with 35 wells affected; in 1984, 90 spills were registered causing contamination of 100 wells [Dickinson, 1987]. Other spills, like the discharge of perchlorethylene from a dry cleaning business in 1986, led to a shutdown of more than 20 groundwater wells in the Fairvale area near Saint John, New Brunswick. Seventy families had to receive their water supplies by trucks for 8 months. It is not known at this moment when the Fairvale aquifer will be usable again. The Fairvale water system had to be connected to another system at a cost of approximately $850,000 [Fairvale, 1986].

All these incidents have led to an increasing concern about groundwater quality and the threat of pollution. This is especially the case in New Brunswick, which relies heavily on this water resource for drinking purposes. It has also been recognized that sufficient information is not readily available on potential pollution sources and the way they might affect groundwater wells in case of a spill.

A project was started in the summer of 1987 by the Water Resource Planning Branch, N.B. Department of Municipal Affairs and Environment, to carry out an inventory of existing potential pollution sources. The project also aimed at determining the susceptibility of affected aquifers from which municipal wells draw their water. Together with an evaluation of the replacement costs of wells, a ranking scheme was established to indicate wells needing protection from potential pollution. The actual inventory was carried out by the water supply consulting firm Hydra Ltd., Youngs Cove, New Brunswick, during the second part of 1987 and was completed in February, 1988.

While the above mentioned project was carried out manually, one can perceive employing expert system technology (see chapter 2.3) to perform a similar analysis. A significant amount of knowledge is necessary to establish the ranks. (This expertise can be contained in a classification system as described in Section 2.3.1). A second step towards the automation of the well site ranking would be to retrieve the necessary geographical and other information from a database in an integrated system as it is shown in Figure 3.2.

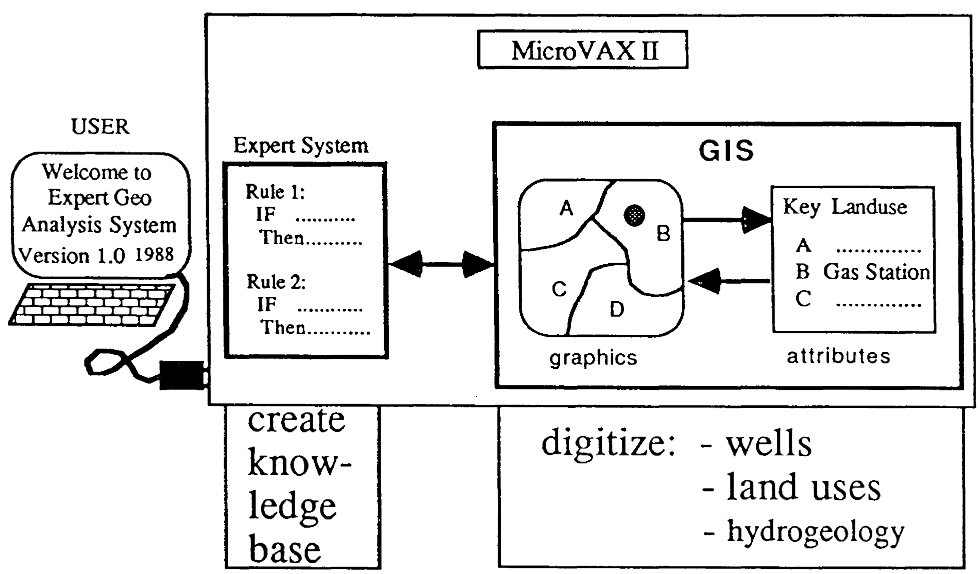

Figure 3.2: Integrated Expert Geo Analysis Systems

3.2. The Process of Ranking Well Sites

A number of methods to classify aquifers with respect to pollution vulnerability have been developed since the potential of groundwater contamination was recognized. One of the first models of classification was developed as early as 1963 by Harry LeGrand.

A fairly detailed method, DRASTIC, was developed by the U.S. Department of Commerce in 1985. All these models help to determine the pollution susceptibility of groundwater using some kind of scoring system for the significant parameters.

The Hydra project goes beyond the scope of most models, by not only determining the pollution susceptibility of wells in a certain area, but also establishing a value of the replacement costs for each individual well, and thus being able to rank comparatively the protection needs for the wells. The method used to rank the wells employs a combination of environmental and economic reasoning. Three types of classification are considered as being important in the ranking of well sites in New Brunswick [Gregory, 1987a]:

1) Pollution susceptibility classification
2) Replacement cost classification
3) Pollution hazard classification

For each of these parameters a scoring system was developed. Depending on the individual scores, a classification value for each factor can be established. The combination of these values determines in which group the well is ranked (see Table 3.1). Group 1 wells, for example, have a high susceptibility and, at the same time, have very high costs of replacement. If potential pollution sources are threatening these wells, they deserve the highest priority for protection against contamination according to this scheme. This priority for attention decreases with increasing well group numbers. Wells within a group are further ranked by the number of potential pollution sources.

Table 3.1: Well Ranking Scheme

	Characteristics	
Well Group	Susceptibility	Replacement Cost
1	high	very high
2	high	high
3	high	moderate
4	moderate	very high
5	moderate	high
6	moderate	moderate
7	low	very high
8	low	high
9	low	moderate

The scoring system represents an organized ranking of the different well sites. This systematic approach can be almost directly translated into the knowledge base for an expert system. The three parameters for the well site ranking can be expressed in three modules of the system. The modules can then be processed together to establish the ranking, or individually to derive, for example, a value for the replacement costs of a well.

3.2.1. Pollution Susceptibility Classification

The susceptibility of a well to contamination is related to the amount of dilution and other attenuation, which occur while infiltrating water follows the shortest path to the operating well [Peters, 1987]. In order to understand this and other pathways, the geology in the immediate area of the well must be known in detail. The compilation and analysis of the hydrogeological data is an extremely complex process.

In most cases, groundwater experts do not have all necessary data at hand to develop an exact model describing the groundwater flow. This can usually only be established after a study over several years using tracing materials in the groundwater flow system.

For the Hydra project, the pollution susceptibility of a well is determined by using a simplified model with three variables [Gregory, 1987a]:

1) Aquifer type (hydrogeological setting)
2) Permeability of the aquifer
3) Length and condition of casing installed in the well

The model was developed considering the availability of information on the municipal wells and aquifer characteristics. The expert assigns different degrees of susceptibility depending on the aquifer and well data. These numbers are added to establish the final class for pollution susceptibility of a well.

Table 3.2: Pollution Susceptibility Classification Scheme

a) Aquifer type:	Score
unconsolidated, semi-confined	0
fractured, consolidated, semi-conf.	1
unconsolidated, unconfined	2
fractured, consolidated, unconfined	3
b) Transmissivity	Score
less than 100 m^2/day	1
100 - 300 m^2/day	2
more than 300 m^2/day	3
c) Casing length	Score
more than 30 m	0
10 m - 30 m	1
less than 10 m	2
d) Casing age	Score
less than 25 years old	0
more than 25 years old	1
e) Casing corrosion	Score
not corroded	0
corroded	1
<u>Pollution Susceptibility Classification</u>	a+b+c+d+e
low susceptibility	1-3
moderate susceptibility	4-6
high susceptibility	7-10

The main sources of necessary data determining the aquifer type are boreholes. A log of soil and rock is recorded when a well is drilled, which provides the data for determining the aquifer type and its permeability.

An aquifer is a permeable deposit which can yield useful quantities of water when tapped by a well [Bowen, 1980]. The main parameters influencing the susceptibility to pollution of an aquifer are its confinement (i.e. whether it has an overlying, impermeable layer of soil or rock) and its development stage (consolidated, fractured, etc.). An extensive confining layer of clay, for example, makes the penetration of a pollutant very unlikely, or at least reduces the potential for contamination significantly.

Many aquifers in New Brunswick appear to be semi-confined. Due to a lack of information, it is often difficult to differentiate between fully and semi-confined aquifers. For the Hydra project, therefore, all aquifers that are not clearly unconfined, are considered semi-confined [Gregory, 1988].

Permeability is the capacity of a rock or soil to transmit water [Bowen, 1980]. Its value is normally expressed in units of metres/day. The rock type and fracturing of the aquifer determines its permeability value. Transmissivity is a parameter commonly used in groundwater hydraulics to characterize an aquifer. Transmissivity is defined by multiplying the coefficient of permeability (also called hydraulic conductivity) with the thickness of the saturated zone of the aquifer. For the Hydra project a value for the aquifer transmissivity and its porosity was determined or estimated. This gives a measure of the ease of transmission of contaminants through the aquifer, and conversely a negative measure of the residence time and opportunity of attenuation of concentration of the contaminants [Gregory, 1988]. The susceptibility of an aquifer is therefore proportional to its transmissivity and porosity.

The two variables, length and condition of casing installed in the well, are also factors in the vulnerability of a specific well. They influence the possibility of a penetration of pollutants originating on the surface around the well. A shallow well will be penetrated more easily by contamination; so will an old well casing that might show signs of deterioration such as cracks and corrosion.

The scoring scheme can be expressed in the rule-based implementation of the pollution susceptibility classification. A typical rule in the knowledge base is:

RULE_12

```
IF   there is evidence of casing.corroded
THEN casing_corrosion.defined is confirmed
 AND 1 is assigned to casing_corrosion.score.
```

3.2.2. Replacement Cost Classification

The second major parameter considered for ranking a groundwater well in the Hydra project is the cost of replacement. If a well underlies a pollution threat, certain protection measures can be applied. Cost factors allow comparative analysis of the value of wells relative to possible costs or economic dis-benefits of protection measures [Gregory, 1988]. They also allow a further refinement of the well ranking: a well with a high replacement cost will have a higher priority for protection against pollution than a well with moderate replacement cost underlying the same pollution threat.

Considering the likely availibility of data, the following parameters were chosen for establishing values for well replacement costs:

1) Well size and yield
2) Difficulty of obtaining an alternate source of supply
3) Distance of alternate source to distribution system

Each of these parameters partially account for the amount a municipality would have to spend if they wanted to replace a well.

Size and yield of a well mainly influence the costs of equipment for and construction of a new well.

Construction includes items such as drilling and building a new pumphouse.

The difficulty of obtaining an alternate groundwater source depends on the favorability of the general hydrogeology of the surrounding area. This factor determines exploration costs to find and develop a new well site.

A replacement well will often lie outside the existing water distribution system of a municipality. It therefore has to be connected by new pipelines. The distance to the existing system is used as an indicator of the costs, which are required for a connecting pipeline.

For each of the above listed parameters, a cost estimation was carried out in the Hydra project. The component costs were not quantified in the study. For an expert system implementation, however, dollar values have to be specified. A scheme was developed for an expert system application based on the Hydra estimates (see Table 3.3). The values given below represent a rough approximation to actual market values (in 1987 dollars). To equip a rule-base with exact values would be very difficult and could be the topic of a separate study.

Table 3.3: Replacement Cost Classification Scheme

		Costs [in Can. $]	
a) Well yield:		Equipment	Construct.
less than 500 m^3/day	10,000	+yield*40	+yield*40
500 - 1000 m^3/day	15,000	+yield*35	+yield*30
more than 1000 m^3/day	20,000	+yield*30	+yield*20
b) Obtaining alternate source:		Prospecting	
not difficult		2,500	
difficult		10,000	
extremely difficult		25,000	
c) Length of connecting pipeline:		Construction	
less than 500 m		10,000 + length*40	
500 - 2000 m		20,000 + length*35	
more than 2000 m		30,000 + length*30	
d) Value of present well:		Equal-replacement factor	
less than $30,000		0.8	
$30,000 - $80,000		1.0	
$80,000 - $150,000		1.2	
more than $150,000		1.5	

Replacement Cost Classification	Total: (a+b+c)*d
moderate replacement cost	less than 150,000
high replacement cost	150,000 - 300,000
very high replacement cost	more than 300,000

The table values represent prices for the construction of a well of average complexity. Considering that the replacement well should be of a comparable standard to the well it is supposed to substitute, an "equal-replacement" factor was introduced in the rule-base of the expert system. The factor adjusts the calculated well replacement costs, considering the sophistication of the well under investigation, which is expressed in a dollar value for that well.

The rule-base, built on the scheme as it is described above, includes rules such as:

RULE_12

```
IF   well.yield is greater or equal to 500
 AND well.yield is less than 1000

THEN well_yield.defined is confirmed
 AND 15,000+(well.yield*35) is assigned to equipment
 AND 15,000+(well.yield*30) is assigned to construction.
```

RULE_14

```
IF   there is evidence of new_well_price.defined
 AND new_well.price is greater than 300,000

THEN very_high_replacement_cost is confirmed
 AND well.replacement is set to extr_difficult.
```

3.2.3. Pollution Hazard Classification

3.2.3.1. Well Protection Area

One of the goals of the well ranking project is to determine the protection needs of wells against contamination. An important step that has to be taken in this evaluation is the determination of an area around the well in which the infiltration of contaminants will affect it.

Delay time is one of the methods used to define a <u>protection area.</u> Delay times are calculated based upon the travel time of water through the aquifer. For the Hydra project, a 10-year delay time was selected. This represents a compromise between the maximum protection in terms of attenuation and dilution of most contaminants, and the least social and economic impact in terms of restrictions posed on the landuse in the protection area of a well [Gregory, 1988].

The derivation of a delay time is an extremely complex process. A detailed description of this process lies beyond the scope of this study. The final report for the Hydra project [Gregory, 1988] describes the derivation methods used for New Brunswick aquifers and a bibliography of publications on this topic.

3.2.3.2. Potential Pollution Sources

Potentially hazardous landuses, including fuel or chemical storage sites, gas stations, or dry cleaning businesses [Gregory, 1987b] were identified and located within the boundaries of the protection area of water supply wells. For the expert system implementation, they are digitized and then retrieved with the help of a geographical information system.

At the current stage of the Hydra project no distinction is made as to what kind of hazard the landuses represent. The existence of one or more potential pollution sources in the protection area is simply considered an unqualified pollution threat. A limited quantification is made by considering the number of detected hazardous landuses in the ranking of well sites. In addition to the survey of specific hazards, values for domestic heating fuel storage, and also septic tanks in areas not serviced by sewer utilities were estimated based on the number of houses in the protection area.

A more sophisticated approach to classification could include parameters such as the toxicity of substances and the amount of potential contaminants in storage. A more accurate model of the effect of a potential spill could thus be developed. This model could also incorporate other factors such as the distance from pollution source to well. The necessary knowledge for this evaluation could again be captured using an expert system, built upon the prototype which was developed for this study.

3.3. Well Site Ranking

With the identification of potential pollution sources, a value for the third parameter for the well site ranking is established. Final ranks can be determined for New Brunswick's groundwater wells, taking into consideration the values derived from pollution susceptibility and well replacement classification. A typical rule, used to rank the wells with an expert system is:

RULE_2

```
IF   there is evidence of high_susceptibility
 AND there is evidence of high_replacement_cost
 AND there is evidence of landuse.hazardous

THEN pollution_threat is confirmed
 AND 2 is assigned to well.group
```

With the evaluation of this or a similar rule the ranking process is completed. (A comprehensive description of the expert system's reasoning processes is given in Appendix II). Individual wells are grouped by the need for protection relative to other wells. This grouping can then be used to establish groundwater management and protection strategies.

CHAPTER 4

EXPERT SYSTEM IMPLEMENTATION

4.1. Implementation Environment

Once it has been decided to build an expert system as a problem solving tool, one has to think about an effective and efficient implementation in the user's application environment. Points to be considered include:

a) the extent and complexity of the problem,

b) the time and money available for the project,

c) the existing hardware and software in the application environment, and

d) the tools available for the implementation.

In the beginning stage of the project, the building of a PROLOG based system was considered. This plan, however, was discarded after it proved to be too time consuming and not efficient enough to develop a prototype from scratch. The system should rather be designed as a general purpose system. Dedicated routines can then enhance the expert program to build a system that is able to solve problems such as the one described in Chapter 3. The implementation of this concept can be achieved by employing an expert system shell (see Section 2.3.5).

The Department of Surveying Engineering, University of New Brunswick, has a geographical information system (GIS), called CARIS[1], residing on a MicroVAX II minicomputer. Since the interaction with the GIS was a major concern of the project, it was desired to find a shell running on the same host. While shells for PCs can be purchased for as low as $400 (U.S.), prices for minicomputer based programs do not start below $5,000 (U.S.) [Gevarter, 1987]. This is a relatively high price, considering the fact that the project was the first of its kind at the Department. One of the objectives was to determine whether further research should be carried out. This had to be done on the basis of a low budget.

Considering these factors, the following solution was chosen: NEXPERT/OBJECT was selected as a shell upon which the system was going to be built. (A brief description of the shell is given in Section 2.3.6.) This program features many capabilities of top-of-the-line shells, while it is significantly less expensive. A demonstration version (max. 20 rules per knowledge base) of the program was purchased to run on the MacIntosh PC. At the same time, NEXPERT, running under VMS on a MicroVAX, was loaned from Digital Equipment Corporation. This combination provided the environment for system implementation and, at

----

[1]CARIS is a trademark of Universal Systems Ltd.

the same time, for thorough software testing prior to a final purchase.

An expert program development tool on a MacIntosh is desirable due to the widespread application of these computers in the Department. Other departmental research activities could thus benefit from the availability of the tool running on this machine. Transporting the knowledge base from there to run on a MicroVAX does not pose any problems due to full compatibility between the knowledge bases under different operating systems. The expert system can run on the MicroVAX in a runtime environment, which does not require dedicated hardware for the graphic interface. Communication routines (see Section 4.4.) link the shell to external programs, like the GIS package, residing on the MicroVAX.

4.2. Non-graphic Interface for a Runtime Environment

Since NEXPERT comes with a graphic interface requiring window capabilities that are not available in the production environment of this project, a line interface had to be developed. This solution also makes the product more transportable. After developing the knowledge base on a VAX workstation or a MacIntosh, it can be run in a non-graphic environment.

The line interface program consists of a control unit

(NXPLine, see flowchart in Figure 4.1 on the next page) and several subprograms, which handle processes that are usually conducted using windows. All routines are written in FORTRAN 77 on a MicroVax II and make frequent use of the NEXPERT sharable images.

The interface can be run with or without a continuous display of the current focus of the inference engine (StExEng). In several subroutine calls, the system is initialized by loading one or more knowledge bases (KBLoad), volunteering zero or more data (DataVol), and/or suggesting a hypothesis (HypoSug). After the initialization, the inference engine is ready to process the knowledge.

If the system needs input from the user, a request is posed by the interface (Question). If a user wants to know why the system needs this information, he or she can ask "WHY?" before responding to the question. The system will then display the rule (StExEng) which needs the piece of information the system is requesting. Different routines for each data type (QBool, QNum, QStr) are used to volunteer the information to the system.

At the end of the expert system run, a summary of the inferred values can be retrieved. The display can be separated by data and hypotheses and output either onto the screen or into a file.

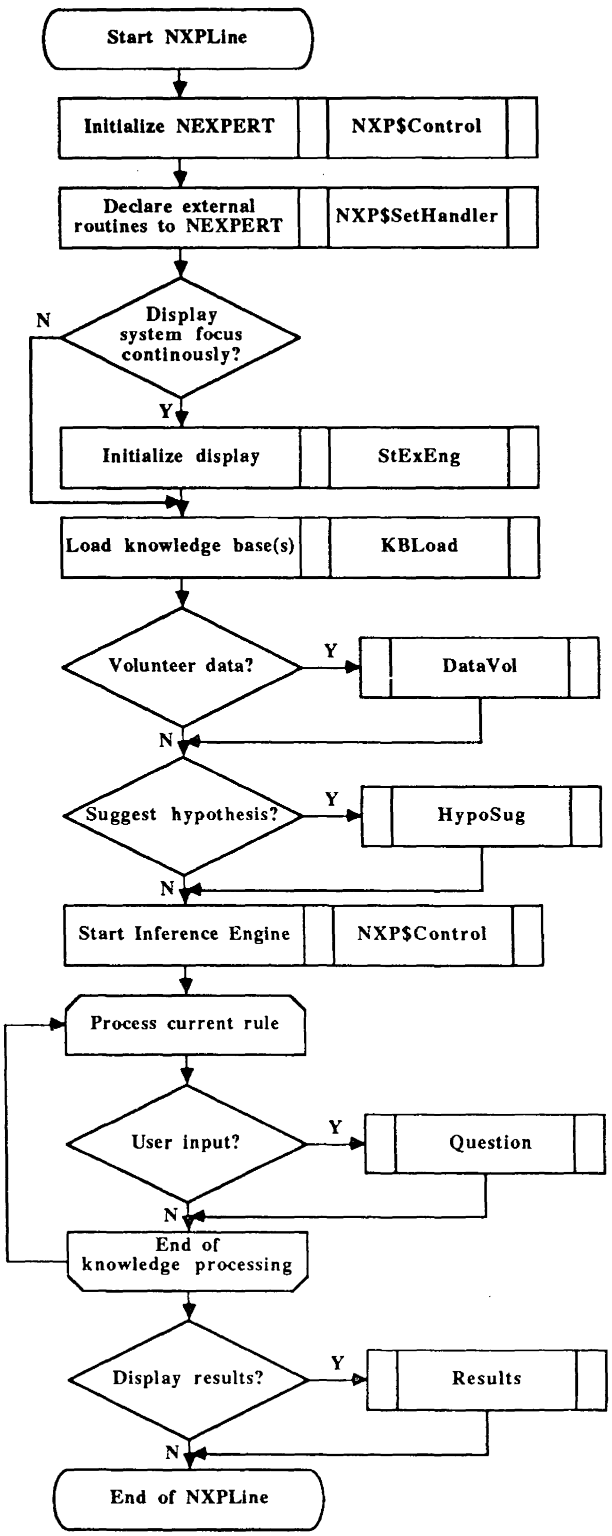

Figure 4.1: Flow Chart of NEXPERT Line Interface

4.3. The Knowledge Base

The categorization of the well site ranking into three subproblems allows for the development of three separate knowledge bases (KB). The restriction of the demonstration system, therefore, did not adversely affect the development. Less than 20 rules were enough to implement the classification scheme for each subproblem given by the expert. The design of a KB, subdivided into separate problem areas, is a common technique in expert system development [Nickerson, 1987b]. If a user is interested only in a part of the problem, the KB can be loaded only partially. This can speed up the knowledge processing significantly, especially for very large knowledge bases.

Forty-two rules proved to be sufficient to implement the expert's well ranking scheme. Figures 4.3, 4.4 and 4.5 display parts of the rule-base network. A complete listing of the rules as they are generated by the development interface on the MacIntosh is given in Appendix I.

At the current development stage of the project, the rules do not allow for the treatment of uncertainties other than accepting that a value is "NOTKNOWN", leading to a conclusion of not known rather than false. NEXPERT allows for various methods of uncertainty treatment, which can be implemented in the knowledge base. The implementation of one, or a combination of several methods, could be a next step in KB development.

In the second part of the knowledge base, the object representation, details are coded about the items the rules deal with. Most objects are defined by specifying them as attributes of rules at the time of the rule-base creation. Some objects or properties of objects, however, have to be added using the object editor. This is, for instance, the case for the number of hazardous landuses in the protection area of a well. This value is of no importance within the rule-base, but is determined by an external routine, which is executed during a system run.

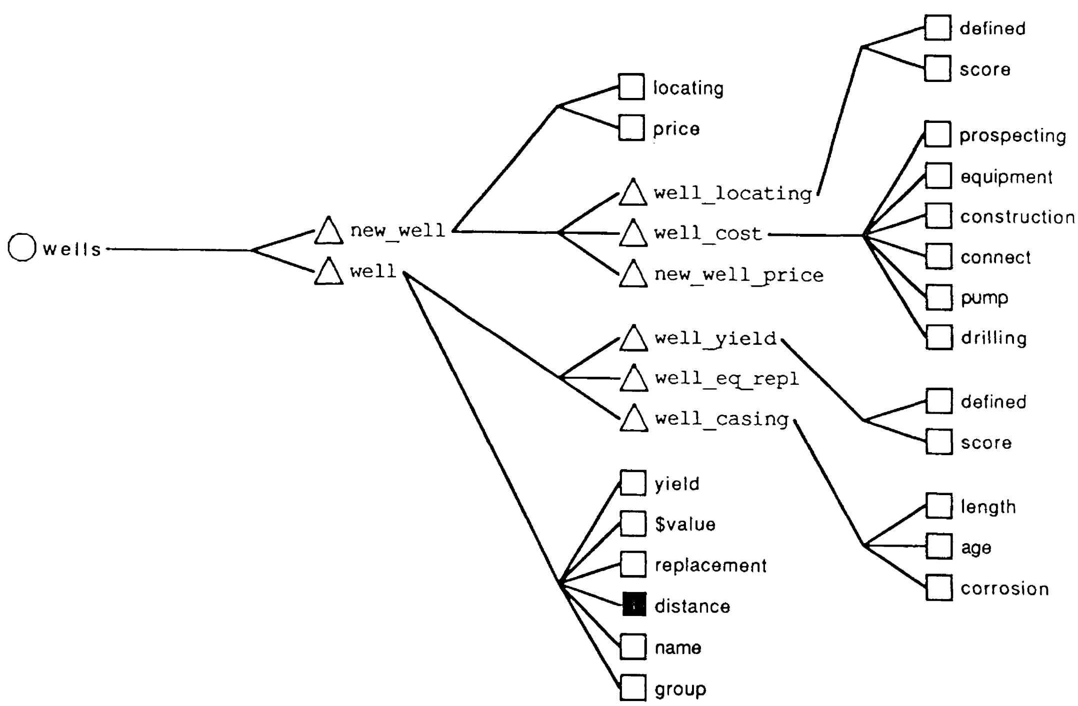

Figure 4.2: Object Representation of Wells

Figure 4.2 shows a part of the object-base as it is displayed by the graphic interface on the MacIntosh. Detailed information about the aquifer and well data, including relationships between classes (represented by a circle), objects and subobjects (triangles), and their properties (squares) are coded in the object representation part of the KB for this project.

The object editor is also used to edit meta-slots of properties (filled squares). In the object-base, one can specify, among other things, where to get certain data, which data to get first in respect to other data, and which question to ask the user, if a data item is needed.

4.3.1. Pollution Susceptibility Module

The first module represents an implementation of the expert's classification scheme (see section 3.2.1) into rules and objects, which comprise the knowledge base (Figure 4.3 displays a part of the rules and their interrelationships). The main hypothesis of this module is the classification of the susceptibility of a groundwater well to pollution. Given this goal, the system backward chains to the point, where it needs input of aquifer and well data. For each data item (object property), one can specify the order of input sources.

Most data for this module are retrieved from an external database (see Section 4.4). If a data item cannot be retrieved, the user is prompted for it. The module contains 16 rules leading to a low, moderate, or high pollution susceptibility value for the well under investigation. This value is used in the well ranking module.

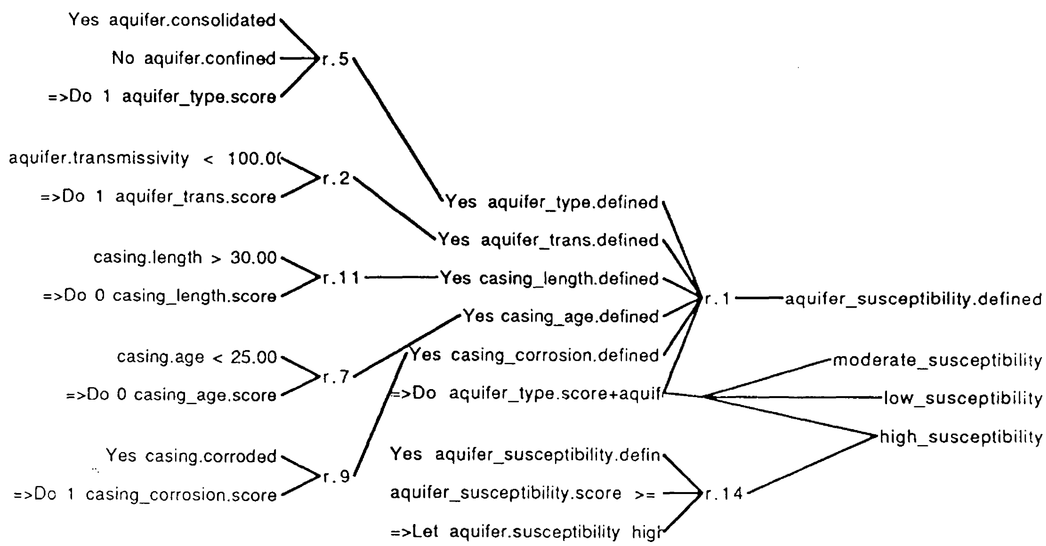

Figure 4.3: Rule Network of Aquifer Susceptibility Module

4.3.2. Well Replacement Module

The second module deals with the replacement costs of a well. The main hypothesis is the establishment of a value for the price of a replacement well. Given this goal, the system backward chains to derive data on the well under investigation and on parameters for a hypothetical replacement well for determining the well's replacement costs.

The module is comprised of 17 rules. It infers the amount of money it would cost to build a replacement well. This value can only be estimated with low accuracy. The classification, therefore, grades wells for moderate, high or very high replacement costs, covering wide ranges of the actual calculated costs. This grading is also used for further processing in the well ranking module.

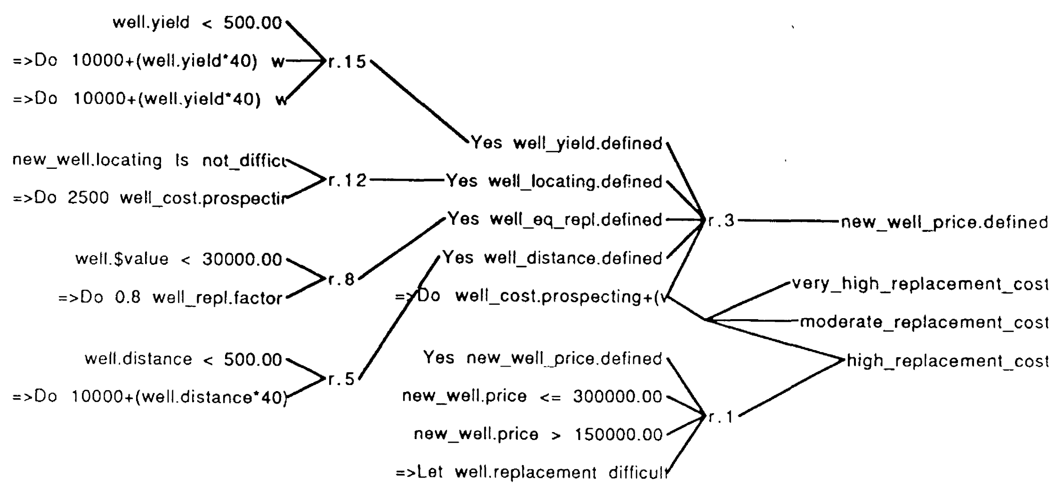

Figure 4.4: Rule Network of Well Replacement Module

4.3.3. Well Ranking Module

The third module does the final ranking of wells into groups of different degrees of pollution threats. Unlike the other units, this is not a stand alone system, which can run on its own. It depends on the results of the two formerly described modules, since its conditions are comprised of the main hypotheses of those.

Figure 4.5: Rule Network of Well Ranking Module

The only hypothesis for the well ranking is the pollution threat. The nine rules rank the well under examination into the nine possible groups. Given the goal, the system backward chains into the other two modules described previously, determines their main hypotheses and ranks the well depending on this evaluation. The third parameter determining the well group, the presence of potential pollution sources, is not derived by backward chaining but by retrieval from the graphical database of the GIS. If the system infers that the well underlies a pollution threat, a map displaying the well and the threatening pollution sources is brought onto the screen using the graphic interface of the GIS (see Section 4.4).

4.4. Communication with External Routines

4.4.1. Where to Get Information

The representation of knowledge and experience gives an expert system the ability to solve problems which require a significant amount of expertise. The power of the problem solving capabilities is limited by the degree to which the transference of real world expertise into the knowledge base succeeds. The usefulness and applicability of the system, however, is additionally limited by the way the system obtains the necessary data about the problem. The system must have access to the same or similar information systems that provide the necessary data for a human expert. It must know where to get information on the data items it is processing on its reasoning path to a solution.

For the well ranking project, a database containing information about municipal wells was created. The database is comprised of two tables. The first one contains data on each individual well and characteristics of the aquifer it draws upon. The second table holds data on the well fields and their protection area. This database comprises the textual part of the GIS database. The graphical part contains a map of Fredericton, showing municipal wells and their protection areas. The keys linking both components are the names of wells and well fields respectively.

The knowledge engineer using NEXPERT has two possibilities of indicating within the knowledge base where and how to get information. He or she can either specify the order of sources in the meta-slots of a property (see Section 2.3.6), or can define a form of data retrieval in conditions or actions of a rule.

One form of obtaining information is by directly accessing a database management system (DBMS). In order to do that, NEXPERT has to be linked to the DBMS when it is installed. Neuron Data provides direct interfaces to several commercially available DBMSs (e.g. Rdb/VMS[1], ORACLE[2]). When the two systems are linked, one can

----

[1]Rdb/VMS is a trademark of Digital Equipment Corporation.
[2]ORACLE is a trademark of ORACLE Corporation.

specify the "RETRIEVE" operator followed by a query language statement to directly retrieve values from the database. Similarly, the "WRITE" operator updates a database with values derived during the expert system run.

The GIS used for this project, CARIS, uses the relational database management system INGRES[3] to manage its textual data. A NEXPERT interface to INGRES is currently under development at Neuron Data. For this project it was not available and external routines had to be written to interface both systems.

The use of external routines is an alternative form of obtaining data from a DBMS. These routines are invoked by the operator "EXECUTE", followed by the name of the routine and a list of arguments comprised of zero or more NEXPERT attributes, which are to be processed. Additionally, a character string can be passed to the routine. A similar technique can be used to build an interface between NEXPERT and the graphical database of the GIS.

If no source for a data item is specified explicitly, the system defaults to prompting the user to input the value interactively. The default prompt for this interaction can be overridden in the property's meta-slot

----

[3]INGRES is a trademark of Relational Technologies Inc.

to issue a user specified request for information.

4.4.2. NEXPERT Callable Interface

External routines interfacing an expert system to other program packages must communicate with both systems. The communication to NEXPERT is established by calling one of the NEXPERT sharable images as documented in "NEXPERT Callable Interface" [Neuron Data, 1987b]. These images also allow for the development of a runtime environment for the system (see chapter 4.2).

The following routines are most commonly used:

- NXP$Control

  A call to this routine gives an external program control over NEXPERT. It is used to initialize NEXPERT, as well as to start and stop the inference engine.

- NXP$LoadKB/NXP$UnloadKB

  These routines enable an external program to load and unload a knowledge base. In an expert system it is often not necessary to load all rules covering every single subproblem. Some of the subproblems might be irrelevant in the current context of the reasoning process. Some subtasks can be controlled by an external routine, which has to load only the knowledge base pertinent to the task.

- NXP$GetAtomID/NXP$GetAtomInfo

  External routines refer to specific NEXPERT atoms by their ID (see Section 2.3.6). GetAtomID returns this internal ID given the name and type of an atom. Another way of retrieving the ID is to pass the atom as a parameter with the "EXECUTE" operator.

  Once the ID of an atom is known, information about this atom can be retrieved from the knowledge base. This information includes names of atoms, values of properties and the type of these values (boolean, numeric, or string), and hypotheses, conditions and actions of rules.

- NXP$Volunteer

  External routines mainly help providing data needed in the reasoning process of an expert system. These data can be calculated using complex mathematical models or simply retrieved from a database. NXP$Volunteer passes the data determined by the routine to NEXPERT.

- NXP$Suggest

  In the beginning of an expert system run, the system is usually initialized by giving it a goal or hypothesis to work on. But also in the course of the problem solving process, a certain development can suggest the examination of a part of the problem that was not previously considered. NXP$Suggest is used to pass a new hypothesis to NEXPERT. Together with the hypothesis, the priority at which the subproblem is to be investigated relative to other hypotheses in the global database, is passed on to the inference engine.

4.4.3. Interface to DBMS INGRES

INGRES (INteractive Graphics and REtrieval System) is a relational database management system. Data are placed in tables (relations) which are organized into rows (records or tuples) and columns (attributes). To manipulate data, one refers to a particular record by specifying database and table names, and giving a value for one or more attributes. Access to a database can be established by both an interactive query language[1] (via the INGRES Terminal Monitor) and a database programming language within a variety of host languages (via E(mbedded)QUEL or ESQL) [Date, 1986].

For this project, database access had to be established by an external program that also links to NEXPERT. The FORTRAN version of ESQL [RTI, 1987] was chosen for this particular application. A number of "Exec SQL ..."

---

[1]INGRES supports both QUEL = "QUEry Language" and SQL = "Structured Query Language".

statements are included in the FORTRAN code, which also contains the calls to "NXP$..." - subprograms which link to NEXPERT. A precompiler translates the ESQL commands into FORTRAN code, which can then be compiled into object code by a regular FORTRAN compiler.

Figure 4.6 shows a flow chart of the subroutine "GetWell", which links NEXPERT to the INGRES database for data retrieval. "GetWell" is a dedicated routine, developed for the particular problem of retrieving well data. It is executed whenever the expert system needs a value for the consolidation of the aquifer, triggered by a specification in the "Order of Sources" within the meta slot of the property "consolidated". This property is given a higher priority relative to other properties, which determine the aquifer susceptibility. Its retrieval triggers the retrieval of all other well data contained in the database.

After opening the database, the user is prompted to enter the name of the well to be investigated. An SQL statement then selects the appropriate data from the record with the well name as its key. These data are then volunteered to NEXPERT, which uses them to derive a value for the aquifer susceptibility.

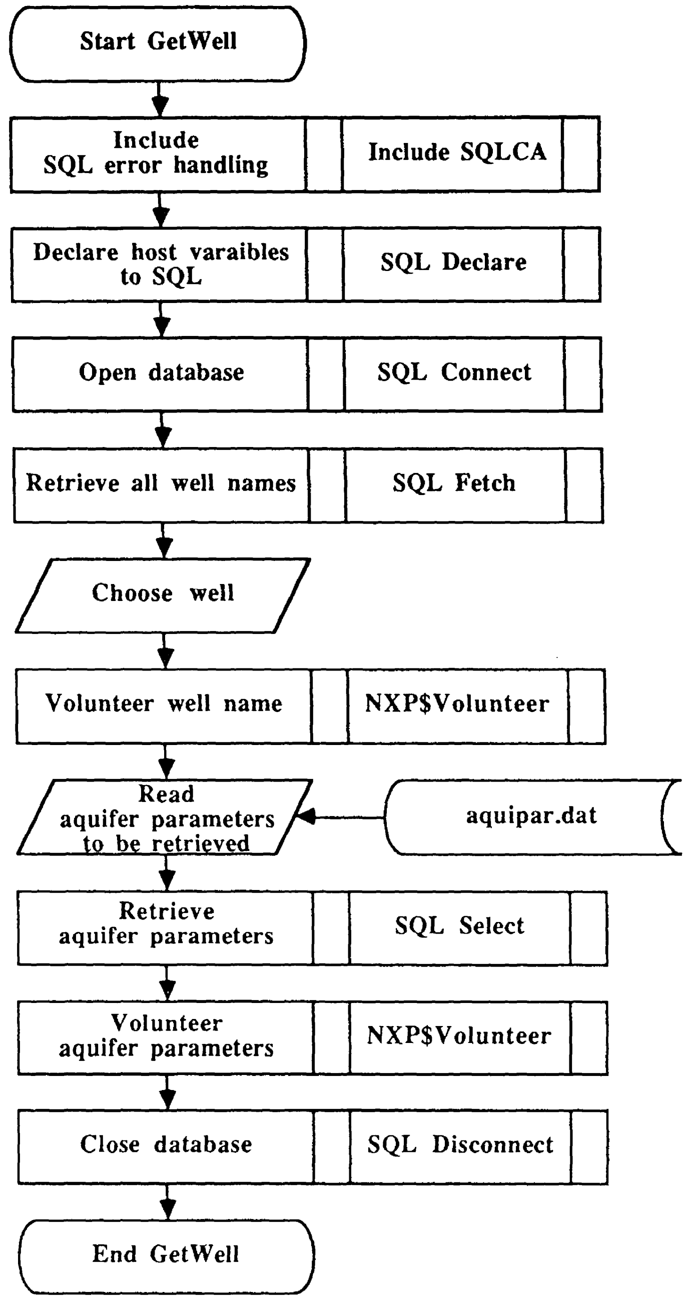

Figure 4.6: Flow Chart of NEXPERT/INGRES Interface

In a future development of the system, this routine could be written as a general purpose retrieval program. NEXPERT could pass the atoms it needs information about, together with database and table names as parameters to the routine. The program then would have to determine the atom names and retrieve the corresponding values from the database using the embedded query statements. A similar routine could allow for updating the database with values inferred by the system. These programs could then act as substitutions of the NEXPERT operators "RETRIEVE" and "WRITE".

4.4.4. Interface to GIS CARIS

CARIS (Computer Aided Resource Information System) is a geographical information system, which manages textual data (attributes) and graphical data separately in two databases. These databases are interconnected by a common key. The use of the attribute database for the well ranking problem is described in the previous section. The graphical database is accessed when the expert system has to determine hazardous landuses within the protection area of a well.

The protection area is represented by a polygon around the well under investigation. A common analysis method using GIS is the point-in-polygon search. Given a polygon, this technique determines all point features of a specified type that are contained in the polygon. The point-in-polygon search can thus find potential pollution sources, which have been digitized, within the boundaries of the well protection area.

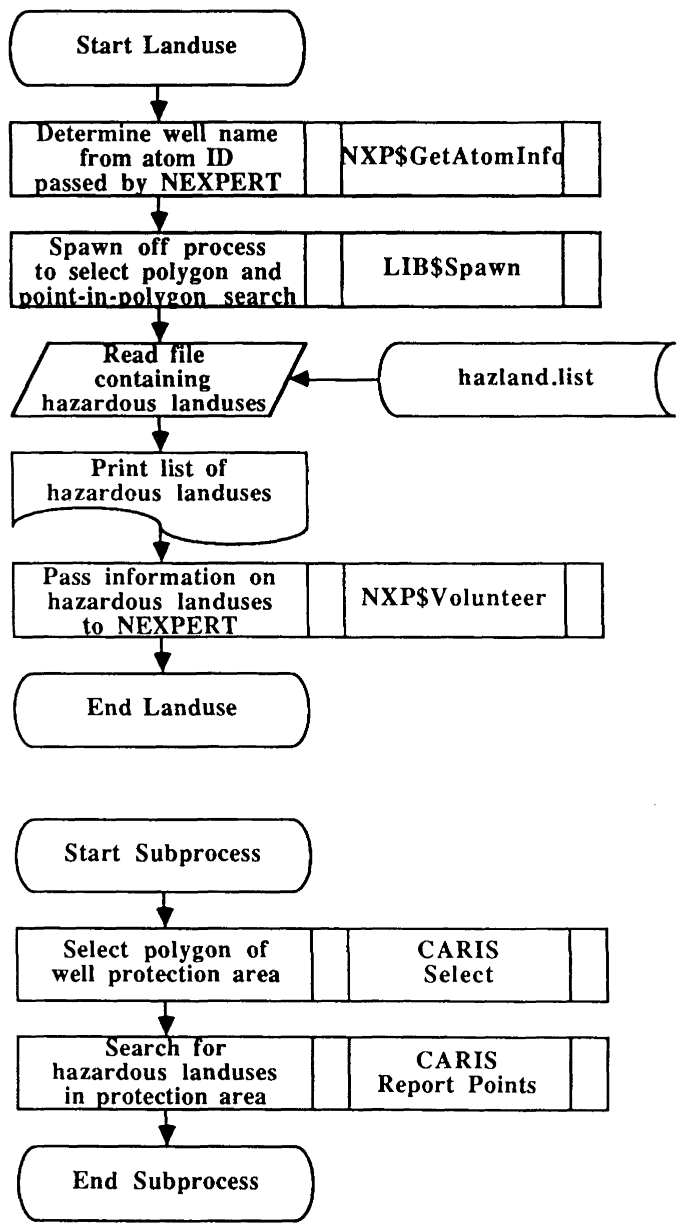

Figure 4.7: Flow Chart of NEXPERT/CARIS Interface

Figure 4.7 shows a flow chart of subroutine "Landuse", which determines hazardous landuses and passes this information on to NEXPERT. The CARIS database is searched using a subprocess, which is spawned off the subroutine. This subprocess uses two CARIS functions to find out about the potential pollution sources: Using the well name as a key, it first selects the polygon which comprises the protection area of the well. Once the protection area is selected, the point-in-polygon search is carried out and the solution sources within the area are written to a file. This file is accessed by the routine "Landuse" after the subprocess is finished, and the information about hazardous landuses passed on to NEXPERT.

The interface described above represents a mechanism, which uses two map analysis capabilities from CARIS to retrieve information which is needed during the path of reasoning of NEXPERT. It is a dedicated routine, which fits the problem of ranking groundwater wells. This routine could be modified to enable the expert system to use the full range of functions available from a GIS. Commands could be passed to the interface and trigger the execution of map analysis function other than the ones used in this implementation. The easiest way to accomplish this is to spawn off the commands as a subprocess. This approach was chosen in this prototype system and demonstrates the possibility of combining both systems. A more advanced solution could be to link to the GIS's data structure for analyzing tasks of greater complexity.

Another use of the GIS can be the employment of its graphic interface. The CARis MANager (CARMAN) offers the capabilities of graphic display of a map together with data retrieval from the attribute database by selecting map features. The routine "ShowMap", triggered by the expert system in the case that a pollution threat is inferred, links NEXPERT to CARMAN. This process is again spawned off the subroutine. Once inside CARMAN, the expert system user can utilize the full interactive capabilities, which are offered by this program, such as retrieving more specific data on the well or pollution sources.

"ShowMap" could also be used to link to the CARis EDitor (CARED). CARED could then be employed to produce a map to support the field work of groundwater experts, or the presentation to other decision makers, which are involved with the problem of groundwater contamination.

CHAPTER 5

SUMMARY, CONCLUSION AND OUTLOOK

5.1. Expert Systems in a GIS Environment

Geographical information systems (GIS) have been successfully applied for two decades. They are an excellent tool for developing and managing an inventory of land resource data due to their ability to integrate large amounts of graphical and textual data. The analytical power of the computer enables GIS users to process and interpret geographical data of much greater complexity than this is possible with conventional maps and data files. Detailed knowledge of system procedures is however necessary to undertake map analysis with a GIS. To make GISs more "intelligent" to better assist the data analysis is therefore an important issue in today's GIS reseach.

Artificial intelligence (AI) is the field of computer science concerned with making machines behave more intelligently. Its most successful branch is expert systems, which can be built to capture the knowledge of human experts and simulate their reasoning processes. Expert systems are considered to be a possible solution of making GIS more intelligent.

Two major issues must be discussed, if one plans to enhance a GIS with an expert system:

1) Where can expert system technology be of advantage in a GIS environment?
2) How can an expert system be integrated into this environment?

This study shows the usefulness of the application of an expert system for data analysis with a GIS. Additionally, several other domains can be identified in which an expert system and techniques of other AI sub-areas could improve the performance of a GIS. Each of these areas represents a key function within a GIS. These functions and possible AI enhancements include:

- Input processing

  Geographical data are collected from different sources. They come in numerous formats and varying accuracies. Data sources can be, among others, hardcopy maps, aerial photographs, field survey data, satellite images and data files from other systems. A GIS should be capable of dealing with these various kinds of input and integrating different forms of data.

  Different types of information must be formatted into a structure suitable for the system. Knowledge about the various input formats provides the means of relating these formats to the system's data structure. This knowledge could be contained in an expert system within an intelligent input processing module of a GIS.

Additionally, some data collection methods provide data in a form that is not suitable for immediate further processing. A map scanner, for example, produces input that requires some preprocessing to recognize map symbols and text, which are simply considered lines upon the completion of the scanning process. Similar problems involve the identification of objects from a satellite image. Pattern recognition techniques can be designed to solve these problems. They can be used to interpret input data and prepare them for storage in the system.

- Output processing

  Map production has been --and still is-- the most common use of GISs. The cartographic quality of GIS map products has been frequently the cause of criticism on system performance. Making good maps, however, is a difficult task. Cartography is a science by itself and has been developing over many centuries.

Several expert system projects have addressed the problem of cartographic quality of GIS maps. AUTONAP for example is a successful expert system implementation for the name placement in maps [Freeman and Ahn, 1984]. Other expert systems, e.g. MAPEX, were developed to automate map generalization [Nickerson and Freeman, 1986]. A common experience of many of these projects is the realization that cartographic knowledge is difficult to formalize and sometimes inconsistent [Robinson and Frank, 1987]. Great efforts must be spent therefore on the knowledge aquisition aspect of the system implementation. Cartographic experts must try to formalize their knowledge together with knowledge engineers, if a cartographic expert system should be successful. During this process it could become apparent that an expert system is not suitable to capture the necessary elements of producing a map satisfying high cartographic standards.

- Database management

The heart of a GIS is its database management system (DBMS). A special property of this DBMS is the integration of textual and graphical data in either a single or a combination of two separate systems. Data retrieval usually can be quite slow due to the large amounts of data that comprise geographical information.

Expert systems can support the DBMS functions. They can enhance the data retrieval mechanism by storing knowledge about data types and their relationships within a database. Query semantics can be used to answer queries in databases more efficiently [Chakravarthy et.al., 1984]. This is particularly true in the selection of data when a query is imprecise [Cromarty et.al., 1984].

Another problem arises with the development of more and more digital databases. The question of efficient use of several distributed databases becomes an issue. An expert system could be part of a query manager to select the appropriate databases to retrieve the requested information.

- User interface

The effective and efficient use of a GIS is only possible after the user has grown very familiar with the system. AI technology could provide GISs with a more intelligent and user friendly interface to simplify its use. An expert system could guide the user through an application. For each GIS function a few rules could capture the knowledge about required and optional user specifications for the function. This knowledge base could then be used not only to support the GIS application but also as a training tool for GIS functions.

A further improvement could be a natural language interface. Users not familiar with the GIS command syntax could thus formulate their requests to the system using natural language. These requests would then be interpreted by the system and translated into functions to be carried out by the GIS.

5.2. Expert System for Well Site Ranking with a GIS

The emphasis of this study was placed on the application of expert system technology for data analysis with a GIS. The ranking of groundwater wells by their needs for protection against contamination was chosen to present such an application. A prototype of a well ranking system was developed during the study to demonstrate the necessary steps for building an expert geographical analysis system.

A generic expert system building tool (expert system shell) was used for system implementation. This shell simplifies the task of system development significantly by providing an interface for the knowledge base creation and an inference engine to process the knowledge. An expert system shell appears to be an excellent instrument for providing a knowledge driven data analysis module for a GIS. Once the shell is linked to the GIS, the user can build separate knowledge bases for each area of application. Expert knowledge used to solve land-related problems can then be duplicated and transported to be accessable at places where an expert is not easily availible. The ease of maintaining the knowledge base allows for uncomplicated and fast changes to keep the knowledge up-to-date.

The prototype developed for this study simulates an analysis carried out manually by a groundwater expert. The knowledge base was built upon the proposals for a well analysis which were obtained from the expert. For the expert's final analysis, however, some of the parameters were changed from those proposed. These changes have been subsequently implemented into the system. The advantage of using an expert system as the problem solving approach became apparent during this change: the system could be changed within one hour without touching any of the analysis routines themselves. The groundwater expert could, in a similar fashion, improve the knowledge base and his ranking model by introducing changes with the rule editor without being concerned with external routines.

This study demonstrates the applicability of expert system technology for data anlysis with a GIS. An expert system shell and several communication routines are sufficient to implement the simulation of a complex analysis problem using GIS functions. The results derived by the system coincide with the conclusions drawn by the human groundwater expert as documented in the final report prepared by Alan Gregory [1988] for Environment Canada.

5.3. Future Research

The research carried out for this project can provide the basis for further work in several areas. Two topics related to the study are suggested for further investigation:

1) Expert system for well site ranking:

The knowledge base for the well site ranking was developed based on a scheme which was used for manual evaluation. No interviews were actually held with the groundwater expert and no iteration took place during the development phase to improve the model step by step. These steps, however, are essential in the creation of a system which is supposed to simulate human expert performance. This was not the objective of the project, but rather the demonstration of the possibility of such a simulation. This possibility is shown by the fact that the system's results coincide with the findings of the expert's study.

Groundwater experts should further develop the system to build a true analysis tool for groundwater management. The advantage of the expert system approach is the flexibility of changing the model and testing its performance in an interactive environment. One aspect of the current model that should be investigated is the incorporation of additional parameters on the potential pollution sources. Data on the amount and toxicity of hazardous substances could be included to give an evaluation of a pollution threat more meaning. As the model is further developed, it should always be considered how a GIS could support the refinements. A close cooperation between groundwater engineers and GIS researchers will be necessary to further investigate this aspect of system development.

While the above proposed research aspect is concerned with the well site ranking using an expert system, another project should be the examination of the expert system itself in more detail. The work with the expert system shell NEXPERT proved the initial assumption of its high sophistication. The shell's functions could only be investigated partially during this study. Ways of dealing with uncertainties, for example, have not been explored at all. This omission makes the implemented model somewhat unrealistic. It also neglects one of the advantages of expert system technology, namely the ability of dealing with uncertainties. NEXPERT offers several possibilities of handling uncertainties which should be explored in a project to refine the model developed in this study.

2) Expert system for data analysis with a GIS:

A number of external routines have been developed for linking the expert system and the GIS used in this project. These routines were designed specifically for the well ranking problem. Research should be carried out towards the development of general purpose interfaces between the systems. This could be done by using a bottom-up approach based on the existing, dedicated programs.

Two routines can establish a basic link to the GIS's database management system for data retrieval and updates respectively. Generic query statements would have to be written. These statements could, together with data items passed by the expert system, build the interface permitting information retrieval for knowledge processing, and database updating with values derived from this process.

A second link is needed to give the expert program access to GIS functions. For this study these functions were simply spawned off as commands from the interfacing routine. This technique was sufficient to prove the usefulness of the system link. A further study should investigate a more direct link to utilize necessary analysis functions. More complex analyses could be carried out by accessing the GIS's data structure, which would be necessary to reason on complicated geometric relations. One aspect of this study would be the investigation of Bobrow's proposition about the inapplicability of expert systems to problems involving complex spatial relations [Bobrow et.al., 1986].

The system developed during this study demonstrates some prospects of a GIS/expert system link. The above proposed project could actually integrate the systems. A fully integrated system would represent a very powerful instrument for supporting decision making processes involving land-related information.

REFERENCES

Barstow, David R. et.al. (1983). "Languages and Tools for Knowledge Engineering." In Hayes-Roth, Waterman, Lenat (eds.) Building Expert Systems. Reading: Addison-Wesley. 283-345.

Bashkow, Theodore R. (1983). "Computer." Colliers Encyclopedia, Vol. 7, 1983 ed: 113-20.

Bobrow, D.G., Mittal, S. and Stefik, M.J. (1986). "Expert Systems: Perils and Promise." Communication of the ACM. Vol.29, No.9. 880-894.

Bowen, Robert (1980). Ground Water. London: Applied Science Publishers Ltd.

Bratko, Ivan (1986). PROLOG: Programming for Artificial Intelligence. Reading: Addison-Wesley.

Brownston, Lee; Farrell, Robert; Kant, Elaine and Martin, Nancy (1985). Programming Expert Systems in OPS5: An Introduction to Rule-based Programming. Reading: Addison-Wesley.

Buchanan, Bruce G. et.al. (1983). "Constructing an Expert System." In Hayes-Roth, Waterman, Lenat (eds.) Building Expert Systems. Reading: Addison-Wesley: 127-67.

Chakravarthy, U.S.; Fishman, D.H. and Minker, J. (1984). "Semantic Query Optimization in Expert Systems and Database Systems." In L. Kerschberg (ed.) Expert Database Systems. Proceedings from the First International Workshop. 659-74.

Coles (1979). Coles Concise English Dictionary. Toronto: Coles.

Cromarty, Andrew S. et.al. (1984). "Distributed Database Considerations in Expert System for Radar Analysis." In L. Kerschberg (ed.) Expert Database Systems. Proceedings from the First International Workshop. 505-24.

Date, C.J. (1986). An Introduction to Database Systems. Vol. 1, 4. Ed. Reading: Addison-Wesley.

Davenport, David C. (1983). "Water Resources." Colliers Encyclopedia, Vol. 23, 1983 ed: 349-57.

Dickinson, Goeff R.E. (1987). Lecture on Groundwater Engineering. University of New Brunwick. Fredericton, 16 Sept. 1987.

"Fairvale Water Contract Sealed." Telegraph Journal [St. John, N.B.] 27 Aug. 1986.

Feigenbaum, Edward A. and Barr, Aaron (1982). The Handbook of Artificial Intelligence. Vol. 1+2. Los Altos: Kaufmann.

Feigenbaum, Edward A. and Cohen, Paul R. (1982). The Handbook of Artificial Intelligence. Vol. 3. Los Altos: Kaufmann.

Freeman, H. and Ahn, J. (1984). "AUTONAP - An Expert System for Automatic Map Name Placement." Proceedings of the First International Symposium on Spatial Data Handling. 554-71.

Gevarter, William B. (1987). "The Nature and Evaluation of Commercial Expert System Building Tools." IEEE Computer. Vol. 20, No. 5: 24-41.

Gray, Peter (1984). Logic, Algebra and Databases. Chichester: Ellis Horwood.

Gregory, Alan (1987a). "Proposed Well Site Ranking." Proposal for project meeting, Hydra Ltd.

----- (1987b). "Preliminary List of Potentially Hazardous Land Uses." Proposal for project meeting, Hydra Ltd.

----- (1988). "Inventory and Review of New Brunswick Municipal Groundwater Supply Areas." Report prepared for Environment Canada, Conservation and Protection, Atlantic Region. Youngs Cove, N.B.: Hydra Ltd.

Hayes-Roth, F., Waterman, D.A. and Lenat, D.B. (eds.) (1983). Building Expert Systems. Reading, Ma.: Addison-Wesley.

Hess, P.J. (1981). "Groundwater Use in Canada." Environment Canada, 1986.

Jackson, Philip C. (1974). Introduction to Artificial Intelligence. New York: Mason/Charter.

Kerschberg, Larry (ed.) (1984). Expert Database Systems. Proceedings of the First Int. Workshop at Kiawah Island, Oct. 84. Menlo Park, Ca: Benjamin Cummings.

Kowalski, Robert (1979). Logic for Problem Solving. New York: North Holland.

Kumara, S., Soyster, A.L. and Kashyap, R.L. (1986). "An Introduction to Artificial Intelligence." Industrial Engineering. Vol.18, No.12: 9-20.

LeGrand, Harry (1964). "System of Evaluation of Contamination Potential of Some Waste Disposal Sites." Journal of the American Water Works Association. Vol. 56, No. 7: 959-74.

Lenat, Douglas et.al. (1983). "Reasoning about Reasoning." In Hayes-Roth, Waterman, Lenat (eds.) Building Expert Systems. Reading: Addison-Wesley. 219-39.

McCarthy, John (1983). "Artificial Intelligence (AI)." Colliers Encyclopedia, Vol. 1, 1983 ed.: 714-16.

McTamaney, Louis S. (1987). "Mobile Robots: Real-Time Intelligent Control." IEEE Expert. Vol.2, No.4: 55-68.

Michalski, R.S.; Carbonell, J.G. and Mitchell, T.M. (1983). Machine Learning. Los Altos: Kaufmann.

Neuron Data (1987a). NEXPERT/OBJECT Fundamentals. Software manual. Palo Alto, Ca.: Neuron Data Inc.

----- (1987b). NEXPERT Callable Interface. Software manual. Palo Alto, Ca.: Neuron Data Inc.

Nickerson, Brad (1987a). "Overview of Expert Systems." Second International Seminar on Trends and Concerns of Spatial Science. U of New Brunswick, Fredericton, N.B.

----- (1987b). Personal consultation. U of New Brunswick, 26 August.

----- (1988). Personal consultation. U of New Brunswick, 15 February.

Nickerson, B.G. and Freeman, H. (1986). "Development of a Rule-based System for Automatic Map Generalization." Proceedings of the Second International Symposium on Spatial Data Handling. 537-56.

Pepper, Don (1987). "AI in Canada Poised for Growth." Computing Canada. Vol.13, No.17: 17-18.

Peters, Lawrence (1987). Personal consultation. U of New Brunswick, 1 September.

Peters, Lawrence (1988). Personal consultation. U of New Brunswick, 15 February.

Prade, H. (1983). "A Synthetic View of Approximate Reasoning." In _Proceedings of the 8th Int. Joint Conference on Artificial Intelligence._ 130-36.

Price, Micheal (1985). _Introducing Groundwater._ London: George Allen & Unwin.

Rich, Elaine (1983). _Artificial Intelligence._ New York: McGraw Hill.

Robinson, Vincent B. and Frank, Andrew U. (1987). "Expert Systems for Geographic Information Systems." _Photogrammetric Engineering and Remote Sensing._ Vol. 53, No. 10, October 1987: 1435-41.

RTI (1987). _INGRES embedded SQL User's Guide._ Software manual. Alamedo, Ca.: Relational Technologies Inc.

Schank, R.C. (1983). "The Current State of AI: One Man's Opinion." _The AI Magazine._ Winter/Spring 1983: 3-8.

Stefik, Mark et.al. (1983). "The Architecture of Expert Systems." In Hayes-Roth, Waterman, Lenat (eds.) _Building Expert Systems._ Reading: Addison-Wesley: 89-126.

Turner, Raymond (1984). _Logics for Artificial Intelligence._ Chichester: Ellis Horwood.

Uhlenbruck, Matthias (1987). "Expert System Application for LIM." Graduate Seminar. U of New Brunswick.

Uhlenbruck, M. and McLaughlin, J.D. (1987). "Expert System Applications for Land Information Management." Proceedings of the _ACSM/ASPRS 1987 Fall Convention_ in Reno, Nv. American Congress on Surveying and Mapping.

U.S. Dept. of Commerce (1985). _DRASTIC: A Standardized System for Evaluating Ground Water Pollution Using Hydrogeological Settings._ National Technical Information Service. Worthington, National Water Well Association.

Weiss, S.M. and Kulikowski, C.A. (1984). _A Practical Guide to Designing Expert Systems._ Totowa, N.J.: Rowman and Allenheld.

Wiig, K. (1985). "Market Trends in AI in the U.S. and Japan." In _Artificial Intelligence: Towards Practical Applications._ Bernhold, T. and Albers, G. (eds.). Amsterdam: Elsevier. 169-80.

Winston, Patrick and Horn, Berthold (1981). _LISP._ Reading: Allison-Wesley.

WRPB (1987). "Groundwater Information Pilot Project." Project definition.

Zadeh, L.A. (1965). "A Theory of Approximate Reasoning." In Hayes, Michie and Mikulich (eds.) _Machine Intelligence._ Vol. 9. New York: John Wiley & Sons: 149-94.

APPENDIX I

Rules Comprising Well Ranking Expert System

a) Aquifer Susceptibility Module (AquiSus2)

b) Well Replacement Module (WellRep2)

c) Well Ranking Module (PollThr2)

RULE : **Rule 1**
If
there is evidence of aquifer_type.defined
And there is evidence of aquifer_trans.defined
And there is evidence of casing_length.defined
And there is evidence of casing_age.defined
And there is evidence of casing_corrosion.defined
Then *aquifer_susceptibility.defined*
is confirmed.
And aquifer_type.score+aquifer_trans.score+casing_length.score+
+casing_age.score+casing_corrosion.score
is assigned to aquifer_susceptibility.score

RULE : **Rule 2**
If
aquifer.transmissivity is less than 100.00
Then *aquifer_trans.defined*
is confirmed.
And 1 is assigned to aquifer_trans.score

RULE : **Rule 3**
If
aquifer.transmissivity is greater than or equal to 100.00
And aquifer.transmissivity is less than 300.00
Then *aquifer_trans.defined*
is confirmed.
And 2 is assigned to aquifer_trans.score

RULE : **Rule 4**
If
aquifer.transmissivity is greater than or equal to 300.00
Then *aquifer_trans.defined*
is confirmed.
And 3 is assigned to aquifer_trans.score

RULE : **Rule 5**
If
there is evidence of aquifer.consolidated
And there is no evidence of aquifer.confined
Then *aquifer_type.defined*
is confirmed.
And 1 is assigned to aquifer_type.score

RULE : **Rule 6**
If
there is no evidence of aquifer.consolidated
And there is no evidence of aquifer.confined
Then *aquifer_type.defined*
is confirmed.
And 0 is assigned to aquifer_type.score

RULE : **Rule 7**
If
casing.age is less than 25.00
Then *casing_age.defined*
is confirmed.
And 0 is assigned to casing_age.score

RULE : **Rule 8**
If
casing.age is greater than or equal to 25.00
Then *casing_age.defined*
is confirmed.
And 1 is assigned to casing_age.score

RULE : **Rule 9**
If
there is evidence of casing.corroded
Then *casing_corrosion.defined*
is confirmed.
And 1 is assigned to casing_corrosion.score

RULE : **Rule 10**
If
there is no evidence of casing.corroded
Then *casing_corrosion.defined*
is confirmed.
And 0 is assigned to casing_corrosion.score

RULE : **Rule 11**
If
casing.length is greater than 30.00
Then *casing_length.defined*
is confirmed.
And 0 is assigned to casing_length.score

RULE : **Rule 12**
If
casing.length is less than or equal to 30.00
And casing.length is greater than 10.00
Then *casing_length.defined*
is confirmed.
And 1 is assigned to casing_length.score

RULE : **Rule 13**
If
casing.length is less than or equal to 10.00
Then *casing_length.defined*
is confirmed.
And 2 is assigned to casing_length.score

RULE : **Rule 14**
If
there is evidence of aquifer_susceptibility.defined
And aquifer_susceptibility.score is greater than or equal to 7.00
Then *high_susceptibility*
is confirmed.
And aquifer.susceptibility is set to high

RULE : **Rule 15**
If
there is evidence of aquifer_susceptibility.defined
And aquifer_susceptibility.score is less than or equal to 3.00
Then *low_susceptibility*
is confirmed.
And aquifer.susceptibility is set to low

RULE : **Rule 16**
If
there is evidence of aquifer_susceptibility.defined
And aquifer_susceptibility.score is less than or equal to 6.00
And aquifer_susceptibility.score is greater than or equal to 4.00
Then *moderate_susceptibility*
is confirmed.
And aquifer.susceptibility is set to moderate

RULE : **Rule 1**
If
there is evidence of new_well_price.defined
And new_well.price is less than or equal to 300000.00
And new_well.price is greater than 150000.00
Then *high_replacement_cost*
is confirmed.
And well.replacement is set to difficult

RULE : **Rule 2**
If
there is evidence of new_well_price.defined
And new_well.price is greater than 300000.00
Then *very_high_replacement_cost*
is confirmed.
And well.replacement is set to extr_difficult

RULE : **Rule 3**
If
there is evidence of well_yield.defined
And there is evidence of well_locating.defined
And there is evidence of well_eq_repl.defined
And there is evidence of well_distance.defined
Then *new_well_price.defined*
is confirmed.
And well_cost.prospecting+(well_cost.construction+well_cost.equipment+
+well_cost.connect)*well_repl.factor is assigned to new_well.price

RULE : **Rule 4**
If
there is evidence of new_well_price.defined
And new_well.price is less than or equal to 150000.00
Then *moderate_replacement_cost*
is confirmed.
And well.replacement is set to not_difficult

RULE : **Rule 5**
If
well.distance is less than 500.00
Then *well_distance.defined*
is confirmed.
And 10000+(well.distance*40) is assigned to well_cost.connect

RULE : **Rule 6**
If
well.distance is greater than or equal to 500.00
And well.distance is less than 2000.00
Then *well_distance.defined*
is confirmed.
And 20000+(well.distance*35) is assigned to well_cost.connect

RULE : **Rule 7**
If
well.distance is greater than or equal to 2000.00
Then *well_distance.defined*
is confirmed.
And 30000+(well.distance*30) is assigned to well_cost.connect

RULE : **Rule 8**
If
well.$value is less than 30000.00
Then *well_eq_repl.defined*
is confirmed.
And 0.8 is assigned to well_repl.factor

RULE : **Rule 9**
If
well.$value is greater than or equal to 30000.00
And well.$value is less than 80000.00
Then *well_eq_repl.defined*
is confirmed.
And 1.0 is assigned to well_repl.factor

RULE : **Rule 10**
If
well.$value is greater than or equal to 80000.00
And well.$value is less than 150000.00
Then *well_eq_repl.defined*
is confirmed.
And 1.2 is assigned to well_repl.factor

RULE : **Rule 11**
If
well.$value is greater than or equal to 150000.00
Then *well_eq_repl.defined*
is confirmed.
And 1.5 is assigned to well_repl.factor

RULE : **Rule 12**
If
new_well.locating is not_difficult
Then *well_locating.defined*
is confirmed.
And 2500 is assigned to well_cost.prospecting

RULE : **Rule 13**
If
new_well.locating is difficult
Then *well_locating.defined*
is confirmed.
And 10000 is assigned to well_cost.prospecting

RULE : **Rule 14**
If
new_well.locating is extr_difficult
Then *well_locating.defined*
is confirmed.
And 25000 is assigned to well_cost.prospecting

RULE : **Rule 15**
If
well.yield is less than 500.00
Then *well_yield.defined*
is confirmed.
And 10000+(well.yield*40) is assigned to well_cost.equipment
And 10000+(well.yield*40) is assigned to well_cost.construction

RULE : **Rule 16**
If
well.yield is greater than or equal to 500.00
And well.yield is less than 1000.00
Then *well_yield.defined*
is confirmed.
And 15000+(well.yield*35) is assigned to well_cost.equipment
And 15000+(well.yield*30) is assigned to well_cost.construction

RULE : **Rule 17**
If
well.yield is greater than or equal to 1000.00
Then *well_yield.defined*
is confirmed.
And 20000+(well.yield*30) is assigned to well_cost.equipment
And 20000+(well.yield*20) is assigned to well_cost.construction

RULE : **Rule 1**
If
there is evidence of high_susceptibility
And there is evidence of very_high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 1 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 2**
If
there is evidence of high_susceptibility
And there is evidence of high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 2 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 3**
If
there is evidence of high_susceptibility
And there is evidence of moderate_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 3 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 4**
If
there is evidence of moderate_susceptibility
And there is evidence of very_high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 4 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 5**
If
there is evidence of moderate_susceptibility
And there is evidence of high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 5 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 6**
If
there is evidence of moderate_susceptibility
And there is evidence of moderate_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 6 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 7**
If
there is evidence of low_susceptibility
And there is evidence of very_high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 7 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 8**
If
there is evidence of low_susceptibility
And there is evidence of high_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 8 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

RULE : **Rule 9**
If
there is evidence of low_susceptibility
And there is evidence of moderate_replacement_cost
And there is evidence of landuse.hazardous
Then *pollution_threat*
is confirmed.
And 9 is assigned to well.group
And Execute showmap(@ATOMID=well.name;@STRING=None;)

APPENDIX II

Example Run of Well Ranking Expert System

Example run of well ranking expert system from a DEC VT220 terminal

User responses to system requests are printed in ***bold italic.***
Comments are printed in *italic*.
For each hypothesis a number indicates its level of backward chaining. **Level 0** hypotheses are the main hypotheses of the system run. For each level of backward chaining the number increases by 1.

```
 NNNN    NN     XXX  XXX     PPPPPPP
NN NN    NN      XXX XXX     PP     PP
NN  NN   NN       XXXXX      PP     PP
NN  NN   NN        XXX       PPPPPPP
NN   NN  NN        XXX       PP
NN   NN  NN       XXXXX      PP
NN    NN NN      XXX XXX     PP
NN    NNNN      XXX  XXX     PP
```

NEXPERT/Object Line Interface

Version 1.0

Do you want a display of the rules during system run [n]? ***y***

Accessible Knowledge Bases:

KBName
aquisusc
wellrepl
pollthrt
aquisus2
wellrep2
pollthr2

Enter names of knowledge base(s); one per line, blank to exit:
aquisus2
wellrep2
pollthr2

Loading knowledge base aquisus2 . . .
Loading knowledge base wellrep2 . . .
Loading knowledge base pollthr2 . . .

Do you want to volunteer some data [n]? ***n***

Do you want to suggest a hypothesis [n]? ***y***

List of possible hypotheses:

Hypothesis	HypNumber
aquifer_susceptibility.defined	1
aquifer_trans.defined	2
aquifer_type.defined	3
casing_age.defined	4
casing_corrosion.defined	5
casing_length.defined	6
high_replacement_cost	7
high_susceptibility	8
low_susceptibility	9
moderate_replacement_cost	10
moderate_susceptibility	11
new_well_price.defined	12
pollution_threat	13
very_high_replacement_cost	14
well_distance.defined	15
well_eq_repl.defined	16
well_locating.defined	17
well_yield.defined	18

Enter Number of Hypothesis to be investigated: ***13***

The system now searches its rule-base for the occurence of the hypotheses pollution_threat. *It starts investigating the first rule found (Pollthr2).*

Current hypothesis: pollution_threat **Level 0**

```
IF
   Yes high_susceptibility and ...
   Yes very_high_replacement_cost and ...
   Yes landuse.hazardous and ...
THEN
   pollution_threat
AND (actions)
   Do 1 well.group and ...
   Execute showmap @ATOMID=well.name;@STRING=None; and ...
```

Value of high_susceptibility is established to be UNKNOWN
Value of very_high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

All conditions in the rule have a value of UNKNOWN. The system tries to evaluate the first condition by backward chaining. The first condition becomes the current hypothesis and a rule is searched for containing this hypothesis (Aquisus2).

Current hypothesis: high_susceptibility **Level 1**

```
IF
   Yes aquifer_susceptibility.defined and ...
   >= aquifer_susceptibility.score 7.00 and ...
THEN
   high_susceptibility
AND (actions)
   Let aquifer.susceptibility high and ...
```

Value of aquifer_susceptibility.defined is established to be UNKNOWN
Value of aquifer_susceptibility.score is UNKNOWN

Again both conditions are UNKNOWN. The system backward chains one more step.

Current hypothesis: aquifer_susceptibility.defined **Level 2**
IF
Yes aquifer_type.defined and ...
Yes aquifer_trans.defined and ...
Yes casing_length.defined and ...
Yes casing_age.defined and ...
Yes casing_corrosion.defined and ...
THEN
aquifer_susceptibility.defined
AND (actions)
Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+
+casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be UNKNOWN
Value of aquifer_trans.defined is established to be UNKNOWN
Value of casing_length.defined is established to be UNKNOWN
Value of casing_age.defined is established to be UNKNOWN
Value of casing_corrosion.defined is established to be UNKNOWN

See above. Third backward chaining.

Current hypothesis: aquifer_type.defined **Level 3**
IF
Yes aquifer.consolidated and ...
No aquifer.confined and ...
THEN
aquifer_type.defined
AND (actions)
Do 1 aquifer_type.score and ...

Value of aquifer.consolidated is established to be UNKNOWN
Value of aquifer.confined is established to be UNKNOWN

At this point the system finds a data item in the UNKNOWN conditions which has the <u>order of sources</u> specified in its meta-slot. This specification triggers the execution of an <u>external program</u> (GetWell) for retrieval from the Wells database.

Opening database wells ...

<u>Choose one of the following wells to investigate:</u>

MAPLE STREET WELL 1
MAPLE STREET WELL 2
MAPLE STREET WELL 3
MAPLE AT MCKEEN WELL
CLIFFE STREET WELL
WILMOT PARK WELL 1
WILMOT PARK WELL 2
WILMOT PARK WELL 3
WILMOT PARK WELL 4
WILMOT PARK WELL 5
GREENWOOD DRIVE WELL 1
GREENWOOD DRIVE WELL 2
MCGLOIN STREET WELL
HIGHLAND AVENUE WELL
CAMERON COURT WELL
DUVAL WELL
RAINSFORD LANE WELL
TOWER ROAD WELL 1
TOWER ROAD WELL 2
KILLARNEY WELL K-1
KILLARNEY WELL K-10

Enter name of well to be investigated: ***DUVAL WELL***

After the execution of the program and retrieval of data on the DUVAL WELL the system goes back to the rule it hypothesis it was investigating before. The data, retrieved from the database, are used for the evaluation of the rule's conditions. The conditions are considered true since the aquifer of the DUVAL WELL is consolidated (Yes) and not confined (No). The previous rule is therefore fired.
After establishing a value for the hypothesis, the system goes one step forward in its reasoning chain to the rule it had been evaluating earlier. The first condition is found to be true from the above evaluation.

```
Current hypothesis: aquifer_susceptibility.defined     Level 2
IF
   Yes aquifer_type.defined  and ...
   Yes aquifer_trans.defined  and ...
   Yes casing_length.defined  and ...
   Yes casing_age.defined  and ...
   Yes casing_corrosion.defined  and ...
THEN
   aquifer_susceptibility.defined
AND (actions)
   Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+
      +casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be true
Value of aquifer_trans.defined is established to be UNKNOWN
Value of casing_length.defined is established to be UNKNOWN
Value of casing_age.defined is established to be UNKNOWN
Value of casing_corrosion.defined is established to be UNKNOWN
```

The second conditions now becomes the hypothesis and the system chains one step back again to find its value.

Current hypothesis: aquifer_trans.defined **Level 3**
IF
< aquifer.transmissivity 100.00 and ...
THEN
aquifer_trans.defined
AND (actions)
Do 1 aquifer_trans.score and ...

Value of aquifer.transmissivity is 95.0

The value for the data item in this rule's condition has already been retrieved by the external program. The rule can thus be immediately evaluated and the system goes again one step forward.

Current hypothesis: aquifer_susceptibility.defined **Level 2**
IF
Yes aquifer_type.defined and ...
Yes aquifer_trans.defined and ...
Yes casing_length.defined and ...
Yes casing_age.defined and ...
Yes casing_corrosion.defined and ...
THEN
aquifer_susceptibility.defined
AND (actions)
Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+
+casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be true
Value of aquifer_trans.defined is established to be true
Value of casing_length.defined is established to be UNKNOWN
Value of casing_age.defined is established to be UNKNOWN
Value of casing_corrosion.defined is established to be UNKNOWN

The third condition becomes the hypothesis now. The system chains backward.

Current hypothesis: casing_length.defined **Level 3**
IF
<= casing.length 30.00 and ...
\> casing.length 10.00 and ...
THEN
casing_length.defined
AND (actions)
Do 1 casing_length.score and ...

Value of casing.length is 21.0

The value has again been already retrieved.

Current hypothesis: aquifer_susceptibility.defined **Level 2**
IF
Yes aquifer_type.defined and ...
Yes aquifer_trans.defined and ...
Yes casing_length.defined and ...
Yes casing_age.defined and ...
Yes casing_corrosion.defined and ...
THEN
aquifer_susceptibility.defined
AND (actions)
Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+ +casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be true
Value of aquifer_trans.defined is established to be true
Value of casing_length.defined is established to be true
Value of casing_age.defined is established to be UNKNOWN
Value of casing_corrosion.defined is established to be UNKNOWN

Current hypothesis: casing_age.defined **Level 3**
IF
< casing.age 25.00 and ...
THEN
casing_age.defined
AND (actions)
Do 0 casing_age.score and ...

Value of casing.age is 10.0

Current hypothesis: aquifer_susceptibility.defined **Level 2**
IF
Yes aquifer_type.defined and ...
Yes aquifer_trans.defined and ...
Yes casing_length.defined and ...
Yes casing_age.defined and ...
Yes casing_corrosion.defined and ...
THEN
aquifer_susceptibility.defined
AND (actions)
Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+ +casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be true
Value of aquifer_trans.defined is established to be true
Value of casing_length.defined is established to be true
Value of casing_age.defined is established to be true
Value of casing_corrosion.defined is established to be UNKNOWN

Current hypothesis: casing_corrosion.defined **Level 3**
IF
Yes casing.corroded and ...
THEN
casing_corrosion.defined
AND (actions)
Do 1 casing_corrosion.score and ...

Value of casing.corroded is established to be UNKNOWN

No value for the corrosion of the casing was retrieved from the database. Since no other source of information is specified in the knowledge base the system prompts the user for a value.

Is the casing corroded ?
Enter value (yes/no/[not known]): ***n***

Current hypothesis: casing_corrosion.defined **Level 3**
IF
No casing.corroded and ...
THEN
casing_corrosion.defined
AND (actions)
Do 0 casing_corrosion.score and ...

Value of casing.corroded is established to be false

Current hypothesis: aquifer_susceptibility.defined **Level 2**
IF
Yes aquifer_type.defined and ...
Yes aquifer_trans.defined and ...
Yes casing_length.defined and ...
Yes casing_age.defined and ...
Yes casing_corrosion.defined and ...
THEN
aquifer_susceptibility.defined
AND (actions)
Do aquifer_type.score+aquifer_trans.score+casing_length.score+casing_age.score+
+casing_corrosion.score aquifer_susceptibility.score and ...

Value of aquifer_type.defined is established to be true
Value of aquifer_trans.defined is established to be true
Value of casing_length.defined is established to be true
Value of casing_age.defined is established to be true
Value of casing_corrosion.defined is established to be true

All conditions are found to be true; the rule is fired. The system now comes another step forward to the rule it had been working on on Level 1.

Current hypothesis: high_susceptibility **Level 1**
IF
Yes aquifer_susceptibility.defined and ...
>= aquifer_susceptibility.score 7.00 and ...

THEN
  high_susceptibility
AND (actions)
  Let aquifer.susceptibility high and ...

Value of aquifer_susceptibility.defined is established to be true
Value of aquifer_susceptibility.score is 3.0

Only the first condition is found to be true and the hypothesis is established to be false. The system goes one more step forward to Level 0 (Pollthr2).

Current hypothesis: pollution_threat **Level 0**
IF
  Yes high_susceptibility and ...
  Yes very_high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 1 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of high_susceptibility is established to be false
Value of very_high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

Since the first condition is false, the rule is not further considered. The system evaluates the next rule with pollution_threat as its hypothesis.

Current hypothesis: pollution_threat **Level 0**
IF
  Yes high_susceptibility and ...
  Yes high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 2 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of high_susceptibility is established to be false
Value of high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

This rule is also not fired and the system goes on evaluating rules with the pollution_threat as their hypothesis.

Current hypothesis: pollution_threat **Level 0**
IF
  Yes high_susceptibility and ...
  Yes moderate_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat

AND (actions)
  Do 3 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of high_susceptibility is established to be false
Value of moderate_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

Current hypothesis: pollution_threat **Level 0**
IF
  Yes moderate_susceptibility and ...
  Yes very_high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 4 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of moderate_susceptibility is established to be UNKNOWN
Value of very_high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

The pollution susceptibility is also not found to be moderate and the next set of rules is discarded.

Current hypothesis: pollution_threat **Level 0**
IF
  Yes moderate_susceptibility and ...
  Yes high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 5 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of moderate_susceptibility is established to be false
Value of high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

Current hypothesis: pollution_threat **Level 0**
IF
  Yes moderate_susceptibility and ...
  Yes moderate_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 6 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of moderate_susceptibility is established to be false
Value of moderate_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

The next rules have a low susceptibility in their conditions.

Current hypothesis: pollution_threat **Level 0**
IF
  Yes low_susceptibility and ...
  Yes very_high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 7 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of low_susceptibility is established to be UNKNOWN
Value of very_high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

The system backward chains and evaluates the rule with low_susceptibility as its hypothesis. The conditions are found to be true and a susceptibility value for the well is established (Aquisus2).

Current hypothesis: low_susceptibility **Level 1**
IF
  Yes aquifer_susceptibility.defined and ...
  <= aquifer_susceptibility.score 3.00 and ...
THEN
  low_susceptibility
AND (actions)
  Let aquifer.susceptibility low and ...

Value of aquifer_susceptibility.defined is established to be true
Value of aquifer_susceptibility.score is 3.0

Current hypothesis: pollution_threat **Level 0**
IF
  Yes low_susceptibility and ...
  Yes very_high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 7 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of low_susceptibility is established to be true
Value of very_high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

The first condition is found to be true. This concludes the evaluation of the first expert system module. The second condition causes the system to backward chain into the second module to establish the costs for a replacement well (Wellrep2).

Current hypothesis: very_high_replacement_cost **Level 1**

IF
 Yes new_well_price.defined and ...
 > new_well.price 300000.00 and ...
THEN
 very_high_replacement_cost
AND (actions)
 Let well.replacement extr_difficult and ...

Value of new_well_price.defined is established to be UNKNOWN
Value of new_well.price is UNKNOWN

Second backward chaining step.

Current hypothesis: new_well_price.defined **Level 2**

IF
 Yes well_yield.defined and ...
 Yes well_locating.defined and ...
 Yes well_eq_repl.defined and ...
 Yes well_distance.defined and ...
THEN
 new_well_price.defined
AND (actions)
 Do well_cost.prospecting+(well_cost.construction+well_cost.equipment+
 +well_cost.connect)*well_repl.factor new_well.price and ...

Value of well_yield.defined is established to be UNKNOWN
Value of well_locating.defined is established to be UNKNOWN
Value of well_eq_repl.defined is established to be UNKNOWN
Value of well_distance.defined is established to be UNKNOWN

Third backward chaining step.

Current hypothesis: well_yield.defined **Level 3**

IF
 >= well.yield 1000.00 and ...
THEN
 well_yield.defined
AND (actions)
 Do 20000+(well.yield*30) well_cost.equipment and ...
 Do 20000+(well.yield*25) well_cost.construction and ...

Value of well.yield is 1008.0

The well yield has also been retrieved already from the wells database.

Current hypothesis: new_well_price.defined **Level 2**
IF
Yes well_yield.defined and ...
Yes well_locating.defined and ...
Yes well_eq_repl.defined and ...
Yes well_distance.defined and ...
THEN
new_well_price.defined
AND (actions)
Do well_cost.prospecting+(well_cost.construction+well_cost.equipment+ +well_cost.connect)*well_repl.factor new_well.price and ...

Value of well_yield.defined is established to be true
Value of well_locating.defined is established to be UNKNOWN
Value of well_eq_repl.defined is established to be UNKNOWN
Value of well_distance.defined is established to be UNKNOWN

Current hypothesis: well_locating.defined **Level 3**
IF
Is new_well.locating not_difficult and ...
THEN
well_locating.defined
AND (actions)
Do 2500 well_cost.prospecting and ...

Value of new_well.locating is UNKNOWN

This value is UNKNOWN and has to be provided interactively by the system user.

What is the locating of new_well ?

Choose one of the following choices for new_well.locating
Choice No. 0 = difficult
Choice No. 1 = extr_difficult
Choice No. 2 = not_difficult
Choice No. 3 = not known
Choice No. 4 = why?

Enter a choice number: ***0***

Current hypothesis: well_locating.defined **Level 3**
IF
Is new_well.locating difficult and ...
THEN
well_locating.defined
AND (actions)
Do 10000 well_cost.prospecting and ...

Value of new_well.locating is difficult

```
Current hypothesis: new_well_price.defined          Level 2
IF
   Yes well_yield.defined  and ...
   Yes well_locating.defined  and ...
   Yes well_eq_repl.defined  and ...
   Yes well_distance.defined  and ...
THEN
   new_well_price.defined
AND (actions)
   Do well_cost.prospecting+(well_cost.construction+well_cost.equipment+
      +well_cost.connect)*well_repl.factor new_well.price and ...

Value of well_yield.defined is established to be true
Value of well_locating.defined is established to be true
Value of well_eq_repl.defined is established to be UNKNOWN
Value of well_distance.defined is established to be UNKNOWN


Current hypothesis: well_eq_repl.defined            Level 3
IF
   < well.$value 30000.00 and ...
THEN
   well_eq_repl.defined
AND (actions)
   Do 0.8 well_repl.factor and ...

Value of well.$value is UNKNOWN


What is the $value of well ?

Enter value [not known]: 100000


Current hypothesis: well_eq_repl.defined            Level 3
IF
   >= well.$value 150000.00 and ...
THEN
   well_eq_repl.defined
AND (actions)
   Do 1.5 well_repl.factor and ...

Value of well.$value is 100000.0


Current hypothesis: well_eq_repl.defined            Level 3
IF
   >= well.$value 80000.00 and ...
   < well.$value 150000.00 and ...
THEN
   well_eq_repl.defined
AND (actions)
   Do 1.2 well_repl.factor and ...
```

Value of well.$value is 100000.0

Current hypothesis: new_well_price.defined **Level 2**

IF
  Yes well_yield.defined and ...
  Yes well_locating.defined and ...
  Yes well_eq_repl.defined and ...
  Yes well_distance.defined and ...
THEN
  new_well_price.defined
AND (actions)
  Do well_cost.prospecting+(well_cost.construction+well_cost.equipment+
    +well_cost.connect)*well_repl.factor new_well.price and ...

Value of well_yield.defined is established to be true
Value of well_locating.defined is established to be true
Value of well_eq_repl.defined is established to be true
Value of well_distance.defined is established to be UNKNOWN

Current hypothesis: well_distance.defined **Level 3**

IF
  < well.distance 500.00 and ...
THEN
  well_distance.defined
AND (actions)
  Do 10000+(well.distance*40) well_cost.connect and ...

Value of well.distance is UNKNOWN

What is the distance from the new well to the water system?

Enter value [not known]: *400*

Current hypothesis: new_well_price.defined **Level 2**

IF
  Yes well_yield.defined and ...
  Yes well_locating.defined and ...
  Yes well_eq_repl.defined and ...
  Yes well_distance.defined and ...
THEN
  new_well_price.defined
AND (actions)
  Do well_cost.prospecting+(well_cost.construction+well_cost.equipment+
    +well_cost.connect)*well_repl.factor new_well.price and ...

Value of well_yield.defined is established to be true
Value of well_locating.defined is established to be true
Value of well_eq_repl.defined is established to be true
Value of well_distance.defined is established to be true

All conditions are now found to be true and the system goes one more step forward.

Current hypothesis: very_high_replacement_cost **Level 1**
IF
Yes new_well_price.defined and ...
> new_well.price 300000.00 and ...
THEN
very_high_replacement_cost
AND (actions)
Let well.replacement extr_difficult and ...

Value of new_well_price.defined is established to be true
Value of new_well.price is 157728.0

The costs for a replacement well have been established but they are not found to be very high.

Current hypothesis: pollution_threat **Level 0**
IF
Yes low_susceptibility and ...
Yes very_high_replacement_cost and ...
Yes landuse.hazardous and ...
THEN
pollution_threat
AND (actions)
Do 7 well.group and ...
Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of low_susceptibility is established to be true
Value of very_high_replacement_cost is established to be false
Value of landuse.hazardous is established to be UNKNOWN

Since the replacement costs are not very high this rule is also discarded and the next one evaluated.

Current hypothesis: pollution_threat **Level 0**
IF
Yes low_susceptibility and ...
Yes high_replacement_cost and ...
Yes landuse.hazardous and ...
THEN
pollution_threat
AND (actions)
Do 8 well.group and ...
Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of low_susceptibility is established to be true
Value of high_replacement_cost is established to be UNKNOWN
Value of landuse.hazardous is established to be UNKNOWN

The system backward chains one step to find out whether the replacement costs are high.

Current hypothesis: high_replacement_cost **Level 1**
IF
  Yes new_well_price.defined and ...
  <= new_well.price 300000.00 and ...
  > new_well.price 150000.00 and ...
THEN
  high_replacement_cost
AND (actions)
  Let well.replacement difficult and ...

Value of new_well_price.defined is established to be true
Value of new_well.price is 157728.0

The costs are found to be high. This concludes the second system module. The first two conditions in the following rule are determined to be true.

Current hypothesis: pollution_threat **Level 0**
IF
  Yes low_susceptibility and ...
  Yes high_replacement_cost and ...
  Yes landuse.hazardous and ...
THEN
  pollution_threat
AND (actions)
  Do 8 well.group and ...
  Execute showmap @ATOMID=well.name;@STRING=None; and ...

Value of low_susceptibility is established to be true
Value of high_replacement_cost is established to be true
Value of landuse.hazardous is established to be UNKNOWN

The evaluation of the third condition causes the system to execute another external program. This program accesses GIS functions to determine potential pollution sources in the protection area of the DUVAL WELL.

Searching for potential pollution sources in vicinity of DUVAL WELL

Selecting polygons ...

In file MAPS:FRED
All themes
In table WELL_FIELDS in database WELLS
where
 c.field_name = 'DUVAL*'
Number of polygons selected is 1
Select complete

In a first step the protection area of the DUVAL WELL is selected.

Northing	Easting	Point Key
679108.000	5092607.000	GAS_AB5
678766.000	5092626.000	AUTO_REP15
678644.000	5091963.000	AN_BARN5

3 potential pollution sources found in vicinity of DUVAL WELL

The point-in-polygon search reveals three potential pollution sources in the protection area. This proves the third condition to be true and the previously displayed rule is fired. The Level 0 hypothesis is thus evaluated and the expert sytem is at the end of its run.
The establishing of an existing polltuion threat triggers now the possibility of the display of the digital map to the user. The VT220 terminal has no graphic capabilities and the option is not used in this example run.

Do you want a graphic display of the vicinity of the well [n]? ***n***

The system finally prompts the user for a display of the results of the evaluation.

Do you want to see a display of some results [y]? ***y***

Do you want the display in a file [n]? ***n***

Choose one of the following categories:
1 Hypothesis(goal)
2 Data

Enter number of categorie: ***2***

Do you want to see a display of all results (y/n)? ***y***

The display module simply dumps the requested results onto the screen or into a file.

Knowledge Base(s): aquisus2
wellrep2
pollthr2

Value of aquifer.confined is established to be false
Value of aquifer.consolidated is established to be true
Value of aquifer.susceptibility is low
Value of aquifer.transmissivity is 95.0
Value of aquifer_susceptibility.defined is established to be true
Value of aquifer_susceptibility.score is 3.0
Value of aquifer_trans.defined is established to be true
Value of aquifer_trans.score is 1.0
Value of aquifer_type.defined is established to be true
Value of aquifer_type.score is 1.0
Value of casing.age is 10.0
Value of casing.corroded is established to be false
Value of casing.length is 21.0
Value of casing_age.defined is established to be true
Value of casing_age.score is 0.0
Value of casing_corrosion.defined is established to be true

Value of casing_corrosion.score is 0.0
Value of casing_length.defined is established to be true
Value of casing_length.score is 1.0
Value of high_replacement_cost is established to be true
Value of high_susceptibility is established to be false
Value of landuse.haz_number is 3.0
Value of landuse.hazardous is established to be true
Value of low_susceptibility is established to be true
Value of moderate_replacement_cost is established to be false
Value of moderate_susceptibility is established to be false
Value of new_well.locating is difficult
Value of new_well.price is 157728.0
Value of new_well_price.defined is established to be true
Value of very_high_replacement_cost is established to be false
Value of well.$value is 100000.0
Value of well.distance is 400.0
Value of well.group is 8.0
Value of well.name is DUVAL WELL
Value of well.replacement is difficult
Value of well.yield is 1008.0
Value of well_cost.connect is 26000.0
Value of well_cost.construction is 45200.0
Value of well_cost.equipment is 50240.0
Value of well_cost.prospecting is 10000.0
Value of well_distance.defined is established to be true
Value of well_eq_repl.defined is established to be true
Value of well_locating.defined is established to be true
Value of well_repl.factor is 1.2
Value of well_yield.defined is established to be true

The display of the results concludes the expert system run.