EXPERT SYSTEM FOR WATER SHORTAGE PREPAREDNESS PLANNING

National Technical University of Athens
School of Rural and Surveying Engineering, Remote Sensing Laboratory

Demetre Argialas, Angelos Tzotsos

Abstract:
The Mediterranean region is facing severe problems of drought and water shortage. Especially islands and coastal areas are highly vulnerable to water scarcity. For this reason, this study aimed to devise a comprehensive strategic and operational plan to combat drought and water scarcity in drought-prone areas. Usually, monitoring drought and water shortage demands a group of experts to apply their knowledge to translate the meteorological and hydrological data into information about the seriousness of the water shortage problem. The object of this research was the development of a prototype expert system, that can capture the domain knowledge of the group of experts involved in this study, for a decision-making mechanism to be derived. Various steps for building the expert system included the problem identification, conceptualization, knowledge acquisition, formalization of knowledge, implementation and validation. The expert system was implemented using the NEXPERT OBJECT tool. The effort was successful and allowed the representation of both complex and simple rule structures that are involved in the water shortage problem. The inferential knowledge was expressed in rules through object oriented structures and AND-OR tree hypotheses. Different strategies were developed, and the paths followed by the system, showed that it is very flexible to handle new information and different inference approaches. Future efforts will concentrate on better generalization and deeper knowledge acquisition.

1. Introduction

1.1 Artificial Intelligence

Artificial Intelligence is a science that has defined its goal as making machines to do things that would require human intelligence. Since Artificial Intelligence (AI) was introduced in the early 1970s, the goal of AI scientists has always been to develop computer programs that can think and solve problems as intelligently as human experts.

Intelligence is, by definition, someone's ability to think, understand and learn instead of doing things by instinct or automatically. Thinking is the process of considering a problem or conceiving an idea. The above definitions motivated scientists and engineers to try to build systems that are capable of learning, solving problems or making decisions in order to be called “intelligent”. The questions about how the human brain works and whether this process can be simulated by a machine have remained unresolved for years, but some of the systems produced by AI researchers were very successful, showing that intelligent behavior can be accomplished by computers.

The problems that humans solve in their day-to-day life are of a wide variety in different domains. Though the domains and the methods differ, AI technology provides a set of formalisms to represent the problems as well as the techniques for solving them.

One of the results of research in the area of artificial intelligence has been the development of techniques which allow the modeling of information at higher levels of abstraction. These techniques are embodied in languages or tools which allow the development of programs that closely resemble human logic in their implementation and are therefore easier to develop and maintain. These programs, which emulate human expertise in well-defined problem domains, are called expert systems. The availability of expert system tools, such as CLIPS, has greatly reduced the effort and cost involved in developing an expert system.

1.1 Knowledge-Based Expert Systems

In the 70's it was accepted that to make a machine solve an intellectual problem, one had to know the solution. In other words, one has to have knowledge about a domain. Knowledge is a theoretical or practical understanding of a subject or a domain. Knowledge is also the sum of what is currently known. The persons who possess the knowledge are called experts. Anyone can be considered a domain expert if he or she has a deep knowledge of both facts and rules and great practical experience in a particular domain. The area of the domain may be limited.

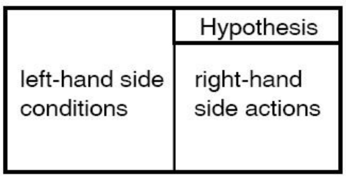
A machine is thought intelligent if it can achieve human-level performance in some cognitive task. To build an intelligent machine, one has to capture, organize and use human expert knowledge in some problem area. Expert systems are computer programs that use domain-specific knowledge to emulate the reasoning process of human experts. An Expert System is not called a program, but a system, because it encompasses several different components, such as knowledge bases, inference mechanisms, explanation facility etc. It was not until the late 1970s that AI scientists realized that the problem-solving power of a computer program mainly derives from the knowledge it possesses rather than the inference mechanism it employs.

The realization that the problem domain for intelligent machines had to be sufficiently restricted marked a major paradigm-shift in AI from general-purpose to domain-specific, knowledge-intensive methods. This led to the development of Expert Systems. Expert Systems use human knowledge and expertise in the form of specific rules and are distinguished by the clean separation of knowledge from the reasoning mechanism. They can also explain their reasoning procedures. But Expert systems can neither learn nor improve themselves through experience. They are individually created and demand large efforts for their development.

The human mental process is internal, and it is too complex to be represented as an algorithm. However, most experts are capable of expressing their knowledge in the form of rules for problem solving. This way, knowledge can be formulated as simple if-then statements, which are called production rules or just rules. Any rule consists of two parts: the IF part, called the antecedent (or the left hand side condition) and the THEN part, called the consequent (or the right hand side action). The basic syntax of a rule is:

IF <antecedent> THEN < consequent>

If the conditions on the left-hand side are satisfied, the Hypothesis of the rule is set to true, the rule is fired, resulting in the performance of actions on the right-hand side of the rule. In general, a rule can have multiple antecedents joined by the keywords AND, OR, or combination of both.



A rule-based expert system has five components: the knowledge base, the user interface, the database, the inference engine and the explanation facilities. The knowledge base contains the domain knowledge useful for problem solving. In a rule-based expert system, the knowledge is represented as a set of rules. The database includes a set of facts used to match against the IF parts of rules stored in the knowledge base. The inference engine carries out the reasoning whereby the expert system reaches a solution. It links the rules with the facts. The explanation facilities enable the user to ask the expert system how a particular conclusion is reached and why a specific fact is needed. An expert system must be able to explain its reasoning and justify its advice, analysis or conclusion. The user interface is the means of communication between a user and the expert system.

Inside the expert system, knowledge is represented by a set of rules and the data is represented by a set of facts about the current situation. The inference engine compares each rule stored in the knowledge base with facts stored in the database. When the IF part of a rule matches a fact, the rule is fired and its THEN part is executed. The fired rule may change the facts by adding a new fact and thus cause another rule to fire. The matching of the IF part of the rule to the facts produces inference chains. There are two kinds of chaining: the forward chaining and the backward chaining.

Forward chaining is the data-driven reasoning that starts from the known data and proceeds forward with that data. Through the inference engine strategy, the appropriate rule is executed. When fired, the rule adds a new fact in the database. Any rule can be executed only once. The match-fire cycle stops when no further rules can be fired. Forward chaining is a technique for gathering information and then inferring from it whatever can be inferred.

Backward chaining is the goal-driven reasoning. In backward chaining, an expert system has a goal (a hypothetical solution or a hypothesis) and the inference engine attempts to find the evidence to prove it. First, the knowledge base is searched so as to find rules that might have the desired solution. Such rules must have the goal in their THEN parts. If such a rule is found and its IF part matches any data in the database, then the rule is fired and the goal is proved. However, this rarely is the case. Thus, the inference engine puts aside the rule it is working with and sets up a new goal, a sub-goal, to prove the IF part of this rule. Then the knowledge base is searched again for rules that can prove the sub-goal. The inference engine repeats the process of stacking the rules until no rules are found in the knowledge base to prove the current sub-goal.

The steps for building an expert system include:

- Problem identification
- Conceptualization
- Knowledge Acquisition
- Formalization of knowledge – Knowledge Representation
- Implementation
- Validation

The first task when solving any problem is the precise definition of the problem in terms of specifications for different situations during the solution process. Knowledge acquisition is the process of extracting, structuring, and organizing knowledge from several sources of knowledge, usually human experts, so that the problem-solving expertise can be captured and transformed into a computer-readable form. Conceptualization is a basis for reasoning: the first hypothesis for a decision or action is the idea that the same or, at least, a similar procedure can yield similar effects within the range of a whole conceptual bunch.

1.2 General Description – Problem Statement

The Mediterranean region is facing severe problems of drought and water shortage. Especially islands and coastal areas are highly vulnerable to water scarcity. For this reason, this study aimed to devise a comprehensive strategic and operational plan to combat drought and water scarcity in drought-prone areas. In the selected study areas, a strategic water shortage mitigation plan was devised based on all possible alternatives which cover a variety of options from demand management to virtual water trade. Moreover, apart from the above proactive water shortage mitigation plan, emergency plans were devised to combat these phenomena or to mitigate the related impacts.

Drought and water shortage levels derive from meteorological and hydrological data. The water balance of an area is calculated as the subtraction of the volumes of water that exist in the storage systems or are gathered through precipitation, from the water demand of the area. In case of a negative water balance, a water shortage is present. But this last case is a worst-case scenario. Even if the water balance is not negative, there must be a proactive plan to alert the administration in order to take measures against the upcoming problem. The administration must be able to know what is the estimated water balance at any time, and what is derived from the hydrological/ meteorological data, as well as what is the interpretation of the data, according to some experts.

Water shortage and drought are monitored annually. But, at any given time, there must be a model able to predict the water availability and the water balance with some accuracy.

Due to heterogeneity of water usage, which depends on local parameters, such a model is very difficult to be formulated. This is why, during this research, three types of land use were considered: Urban, Rural and Mixed.

In order to make a preparedness plan, one has to choose the study area and then decide whether the plan is going to be strategic, or operational. Operational planning is used in both short-term and long-term situations, when immediate action is needed. A strategic plan encapsulates the long term needs in water, along with socioeconomic data so as to derive a viable plan, according to the system's needs.

For example, in the case of an urban system, an operational plan would require the identification of the resources of the study area. The infrastructure, the water sources and their capacity must be recorded, for the water availability to be determined. The interrelations between water storage systems and water demand must be evaluated and then a water balance measurement methodology should be established so that a decision can be reached, whether the problem exists or not. Networks and their current state, as long as population and touristic data must be taken into consideration for evaluation. A monitoring methodology is also needed to evaluate hydrological and meteorological data, and to provide estimation of the water volume to become available in the future. After the estimation of the water balance of the study area, the state of water shortage is computed. Next, the alternative solutions to the water scarcity problem must be evaluated, in accordance with the socioeconomic standards and a multi-criteria analysis is applied to optimize the water usage and distribution. Finally, a public evaluation must take place so that the measures can be viable to the study area.

Due to the fact that, in hydrology, many of the decisions made about drought are empirical, it is difficult to implement an algorithm to simulate the above process. Many parameters are involved in the specification of the monitoring system. Apart from that, the decision-making process is very complex and inexact in nature. This is why an expert system is required against a conventional program so as to objectify the decision and explain the reasoning behind it.

1.3 Objectives of research - Motivation for using KBES

Usually, monitoring drought and water shortage demands a group of experts to apply their knowledge to translate the meteorological and hydrological data into information about the seriousness of the water shortage problem. Such a group of experts was also gathered for the purposes of this research. The major difference is that, apart from the need to develop an operational and a strategic plan/model for the drought-prone areas, there was also a task to participate in an expert system development cycle, where knowledge about the drought and water shortage problem was to be captured and then represented in an intelligent system.

Meteorological and hydrological data are continuously recorded around the globe. In order for a region administration to effectively diagnose an upcoming drought, expert knowledge is required. Since it is a high-cost procedure to have domain experts available around the clock to interpret all the recorded data, it is crucial to be able to make the procedure more automatic and at the same time more objective.

The objective of this research was the development of a prototype expert system, that can capture the domain knowledge of the group of experts involved in the project, for a decision-making mechanism to be derived. The goal was not to have a complete prototype system, but a functional one under a "closed world" assumption. For this purpose, the domain problem was shrunk and simplified in some aspects, but still the development team had as a goal to implement a system that could provide assistance to the end user, so as to plan an operational or strategic plan against water shortage and drought.

2. Methodology

2.1 Problem Identification

The class of problems the expert system was expected to solve was:

- The identification of drought through the monitoring stages and mechanisms. The system should have the knowledge to identify a drought situation and be able to correctly integrate this information with other facts, such as water availability.
- The separation between strategic and operational planning which should be clear in the knowledge base, in order for the same facts and parameters to have different time-frames for planning and consider different possible solutions.
- The conceptual scheme for water sources, consumption, precipitation, drought indices, water balance, risk etc.
- The decision-making for alternative solutions and the sub-goals involved in each solution path.
- The hierarchical structure of alternative solutions as perceived not only by experts but also by the public.
- The scheme for the interrelations of water volumes and the modeling of the water balance system for the study area.
- Heuristics about the empirical knowledge involved in the decision-making.

During the problem identification step, the major decisions about how the problem is fitted best into a conceptual scheme was taken. The system milestones where chosen, as well as the strategies for achieving the main objective.

For the water shortage preparedness planning, it was identified that development could follow two paths. The first one included a backward chaining reasoning with the organization of sub-goals in appropriate knowledge islands. This was meant to simulate a more natural organization and management of the problem. The strategy that the team of experts would follow, was suggested like a workflow, for information to be gathered and interim goals to be reached before making the final decisions about the proactive plan. The second path included a more flexible backward chaining technique, with the ability of forward chaining, so that the end user can decide if he or she wants to provide data early in the execution time, just to gather information about a specific hypothesis, or even to run part of the system in order to learn or understand the knowledge behind the system. In both cases, the same knowledge representation would be used for compatibility, but the rules would be organized in different knowledge islands.

The basic concepts on which the system would focus were:

- The network and infrastructure status, as well as the crucial properties that can provide valuable data and which an expert would definitely want to know before tackling a water shortage problem for an area.
- The hydrologic and satellite indices to be used as monitoring mechanisms.
- The alert thresholds for the above indices, or their derivative concepts.
- The water demand types and how they can be classified to a knowledge scheme.
- The alternative solutions hierarchy and the conditions that can make each of them applicable to the current situation.
- The water availability types and the water sources.
- The interrelation between water volumes of the water system of an area.
- The socioeconomic models that would be available in the system so that the impact of the measures to the public could at some level be taken into consideration.

2.2 Knowledge Acquisition

Knowledge acquisition is the process of extracting, structuring, and organizing knowledge from several knowledge sources, usually human experts so that the problem-solving expertise can be captured and transformed into a computer-readable form.

Knowledge is the most important component of expert systems. Without explicitly represented knowledge, an expert system is no more than a computer program. The process of assimilating the expertise of several experts into an expert system is not easy, particularly when those experts are trained in different disciplines. Differences not only appear in problem-solving strategies employed by each expert, but also appear in what heuristics are applied to solve the problem. The difficulty arises because of the communication barriers among experts and between experts and the knowledge engineer.

The approach used for knowledge acquisition determines both the quality of knowledge and the amount of effort required for its acquisition. The technique selected greatly affects the performance of the expert system and the resources required for its development. The recognition of the importance of knowledge acquisition has resulted in the development of various techniques, methodologies, and tools for automated knowledge acquisition. Some of the techniques for knowledge acquisition include book-level knowledge extraction, interviewing, observation, brainstorming etc.

Interviewing is the most commonly used method. It is used for extracting knowledge from domain experts for expert systems development. There are some difficulties with this technique though. As people become more experienced at performing certain tasks, they become less aware of the cognitive processes involved in their performance. They cannot explicitly describe their reasoning process step by step. Furthermore, there are certain biases in human reasoning.

During the implementation of the proposed system, the interviewing method was used, along with some parts of brainstorming method. Several interview sessions were conducted and the team of experts initially tried to explain the background hydrologic knowledge that the knowledge engineer had to be familiar with for the basic representation of knowledge. During those sessions, the object-oriented representation was implemented and some decisions were made, such as: what kind of reasoning should be followed, what would the degree of freedom for the system asking various questions be, should weights be applied to specific rules so as to emphasize their significance etc.

In the following stages, the basic concepts of the water shortage problem started to evolve into more meaningful knowledge for the system and the first branches of rules were implemented. As the knowledge representation started becoming more complex, several milestones were set in order to test the evolving system and be able to control the functionalities.

In the latest stages of the system development, testing the prototype was a major objective, as well as refining the rules. Some last-minute changes were made, due to the fact that the teams of experts involved in the research project were from different countries, and many rules were also reviewed and then changed according to the changes in other work packages.

All the knowledge that was extracted was formulated and tested in an Expert System Tool called NEXPERT OBJECT.

2.3 Knowledge Engineering Environment

The object-oriented representation structure and the associated inference rules developed earlier were programmed in the expert system tool NEXPERT OBJECT (Neuron Data 1993). This tool provides a graphical representation of both the object and the rule structure as it exists before the program execution or as it unfolds during the dynamic consultation of the expert system. These graphical networks are more declarative than the alternative textual representations and, therefore, they are used in the figures to demonstrate the system operation.

NEXPERT OBJECT provides many representational structures. There are objects and classes to describe the entities in the domain. There are properties which are characteristics of objects and classes. Slots store information about specific objects and classes. There are also meta-slots which describe how the slots behave. Properties can be inherited from a class or object to another class or object. Values can also be inherited from a class or object to another class or object. In addition, NEXPERT OBJECT allows creating objects dynamically during a session. These dynamic objects allow the modeling of a world whose exact structure is not known a priori (for instance how many records are in a database). One can also create dynamic links between objects or classes and other objects or classes to reflect changing relationships during processing. NEXPERT OBJECT supports rules which contain all of the domain knowledge. Rules manipulate the slots as well as the object and class structures. Pattern matching and interpretations allow you to reference objects which are determined at runtime. Thus, one can write generic rules which reason on a set of objects which are determined when the rule is processed.

An object is the smallest chunk of information in the knowledge-based system. It represents any person, place, thing, or idea in the domain for this particular application. People describe their application's world in terms of various objects. A class is merely a grouping or *generalization* of a set of objects. Objects are specific members or *instantiations* of a class. A subclass is a class which represents a subset or *specialization* of another class. It is a class in its own right and has all the characteristics of other classes. Classes can have any number of subclasses or parent classes, or both. We can create a class hierarchy with any number of levels.

2.4 Conceptualization – Knowledge Representation

Knowledge conceptualization and representation aim at uncovering the key concepts of the domain and the relationships between them, as well as conceiving a formal description of knowledge in terms of the primitive concepts and conceptual relations.

For the structural knowledge representation of the water shortage problem, we assume an object-oriented representation structure that uses frames as classes, subclasses, objects, subobjects and slot frames as properties. For the strategic knowledge representation we assume a rule-based inference engine. In the design of the system, we have used the NEXPERT OBJECT (by Neuron Data) expert system tool for its ability to support both a reasoning system and a powerful, object-oriented representation. This makes it a very powerful hybrid system for representing knowledge and building high quality expert systems.

First we identify the need to name and describe by their properties the basic features of our domain:

- The PRODIM_System class for global information storage (Fig 1).
- The Monitoring System class for data processing referring to the monitoring stages of the planning procedure (Fig 1).
- The Location class to define where the current study area of the system is (Fig 1).
- The Drought class to define the various properties of drought and to classify the Drought in accordance with the rules fired and facts provided.
- The Infrastructure class, which defines the status and the properties of the Location in study (Fig 1).
- The Drought Index class, as a basic element of the monitoring system (Fig 2).
- The Water Sources class, for classification of the water volumes that are available to the system
- The Risk class, to define the correlations and conditions that connect risk with drought and water shortage.
- The Solutions class, so that alternative solutions can be recorded, analyzed and proposed by the system.
- The Water Quantity class, which is a class intended to perform various calculations for managing water volumes (either as input or output).
-

Second, we organized the domain classes into class-subclass hierarchies:

- The PRODIM_System entity divided into 3 sub-systems (Agricultural, Urban and Mixed) (Fig 1)
- The Monitoring System divided into 2 sub-systems (Drought Monitoring and Consumption Monitoring) (Fig 1)
- The Infrastructure class is divided into 2 subclasses concerning the network of the study area and the storage facilities (Fig 1).
- The Drought Index is split into 3 classes (1 abstract class included), representing the commonly used hydrologic indices that can be used for drought monitoring (Fig 2).
- The Risk class is divided into scarcity risk and real risk.
- The Water Sources can be either ground, surface or non conventional.
- The Water Quantity class can be specialized into Water Balance, Water Availability and Water Requirements.
- Water Requirements can be specialized into Consumption, Household Use, Touristic Use, Recreation Use, Industrial Use, Agricultural Use and Other Use. This specifications help the system understand where the water volumes are required so as to be able to project the problems arising from each alternative solution suggested.

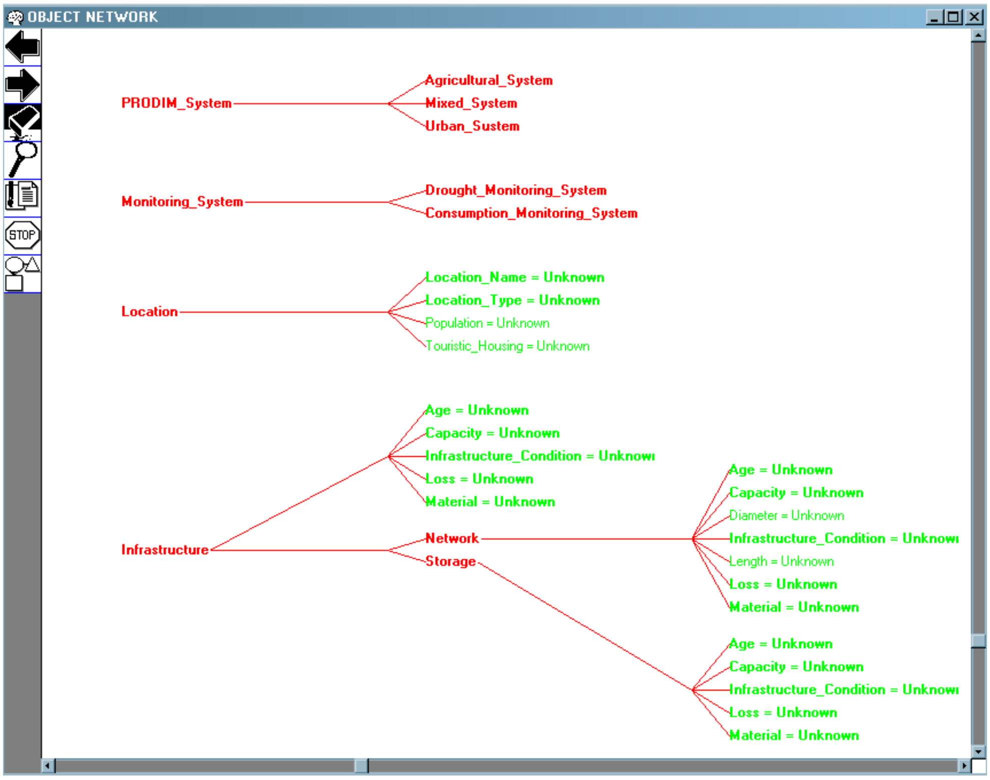


Figure 1. An object-oriented representation for the classes PRODIM System, Monitoring System, Location and Infrastructure

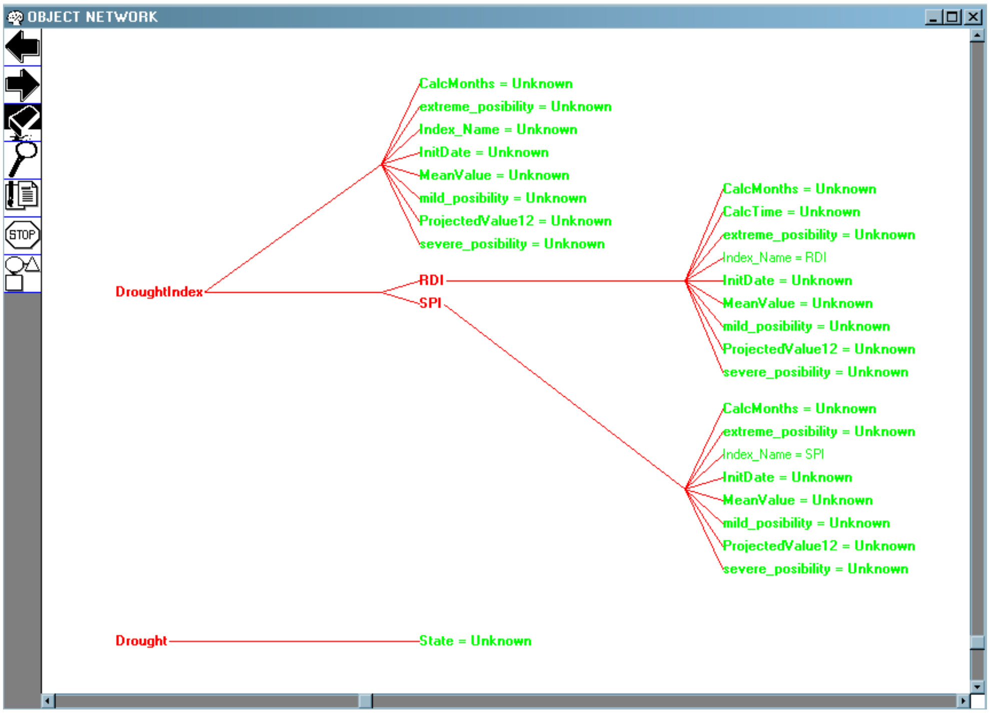
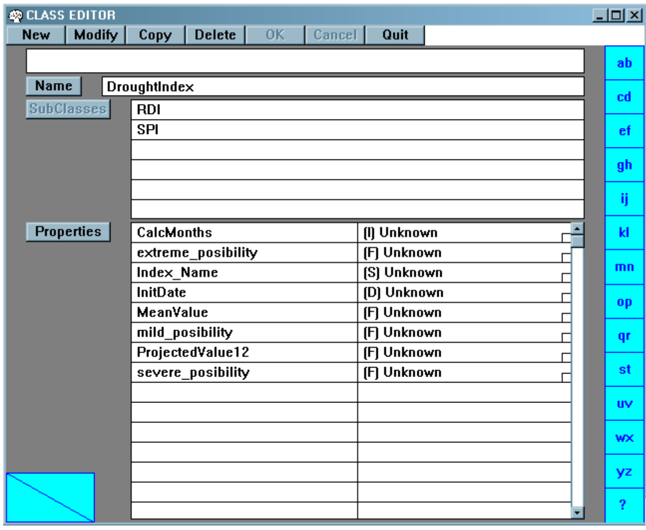


Figure 2. An object-oriented representation for the classes Drought and Drought Index. The classes are represented in red color. The properties and their values are represented in green color.

Third, we defined class members or instances (Fig 4). When a class is to be used for inference, its properties are calculated and the values are used by the inference engine. There are cases where we need to represent that a class was activated many times or that the class has many instances. Then we need to define objects of that class. In other cases, there are variables than do not have an object-oriented representation (e.g. the current task of the system can be a string variable). These variables are represented as single objects without a parent class. All objects can be edited by the object editor of the system (Fig 5).

Subsequently, each class was defined by a set of properties which define the class. Objects and subclasses can obtain their properties dynamically from a particular class through a mechanism called inheritance. Thus, through the class/subclass or class-instance hierarchy, these properties are inherited down each hierarchy so as to be shared by all the members or instances of each class. The properties of the semantics of the water shortage preparedness planning problem were defined so as to reflect the characteristics of each class.

Having defined the classes, subclasses and objects, we use them to describe the water shortage problem reasoning. This reasoning, along with the basic concepts for the above representation, was captured through the process of Knowledge Acquisition.



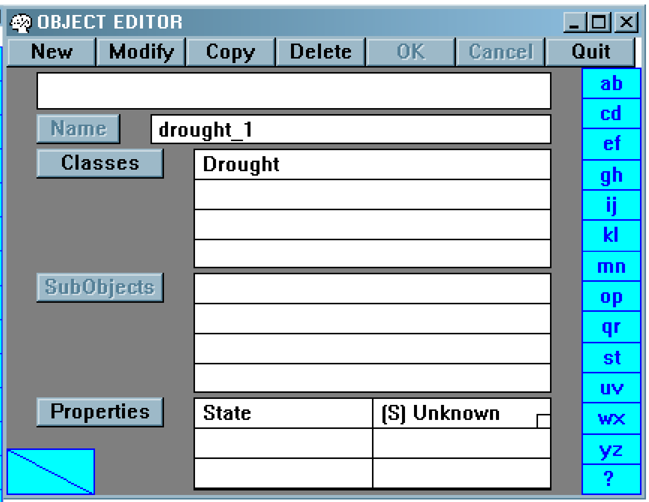


Figure 4. The class editor window for the Drought Index class. The subclasses are defined here, as well as the properties of the class (and the types of the slots)

Figure 5. The object editor. Here, the Drought class instance, object Drought1 is presented. In the subobjects window, the part-of relationships can be declared.

2.5 Formalization of knowledge.

The inferential knowledge about the water shortage problem was expressed in rules. Complex data structures and AND-OR trees were implemented. During execution the rule based tree structure performs a recursive search in the object-oriented representation of the study area. During an execution, the path followed is selected based on the user input, so that the search is limited to a specific problem solving scenario. The aim is to match user input to a class description and fire the proper rules to reach to the appropriate solutions.

In Figure 5, an initial hypothesis of the system is shown. This was suggested by the experts, in order to control from which knowledge island the backward process should begin. This set of 3 rules initializes the system, then declares that the system task is Mixed, Urban or Rural and then goes into the Initial State knowledge island. The first information the system asks this way is the type of study area and the name of this location.

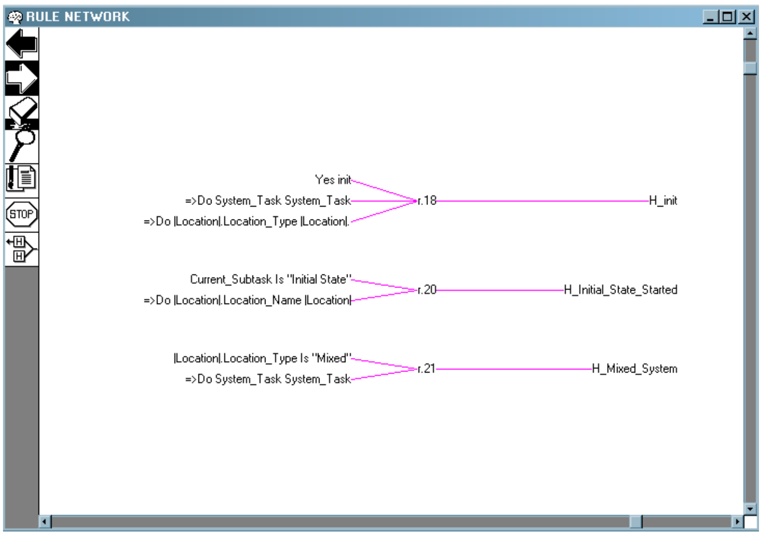


Figure 5: The 3 initial rules of the system.

In Figure 6, the rules that proves that the monitoring process is completed, is shown. This is a rule that makes sure that all conditions and hypotheses needed from this knowledge island were tested and then changes the current task to Risk assessment. Notice that this rule needs for the system to be in operational plan mode and Water Scarcity to be proved true or false. In both cases this rule will become true finally.

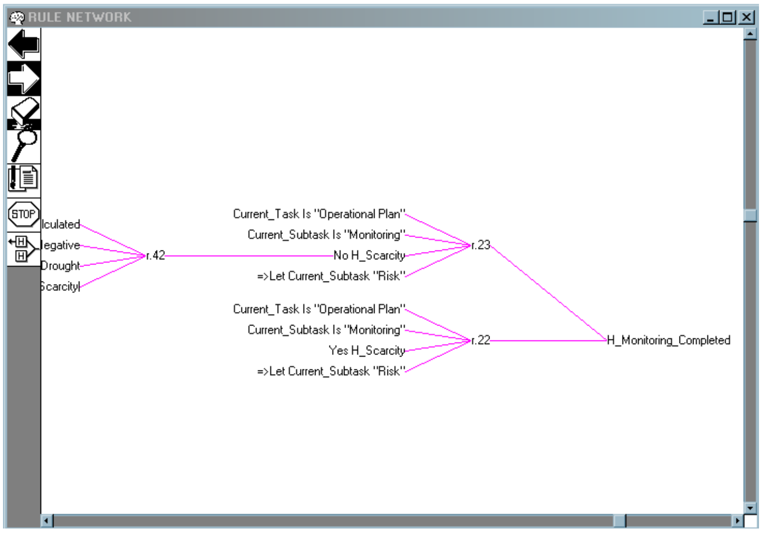


Figure 6: Control over the change of knowledge island is shown

In figure 7, the dynamic creation of an instance of a class is presented. This rule also declares that in order to have water scarcity, drought has to occur and that several other conditions must be met. The creation of a dynamic object is also shown in figure 8, in addition to the computation of water balance.

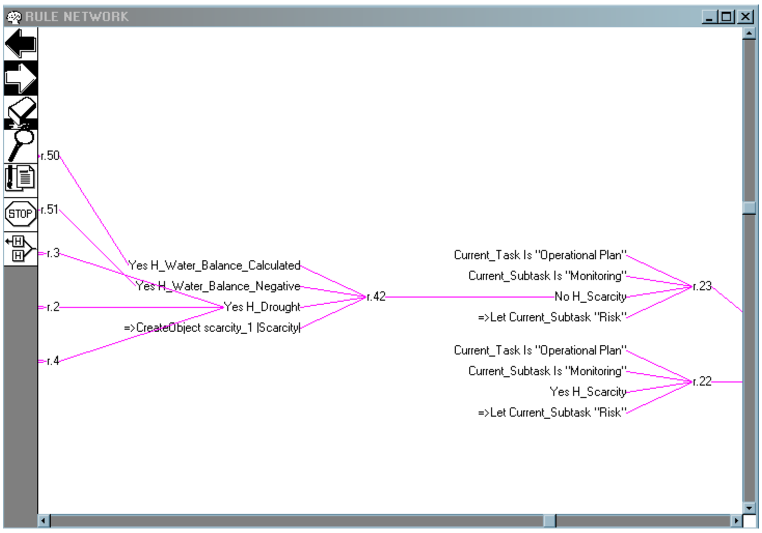


Figure 7: Dynamic creation of a “water scarcity” object once the H_Scarcity hypothesis is set to true.

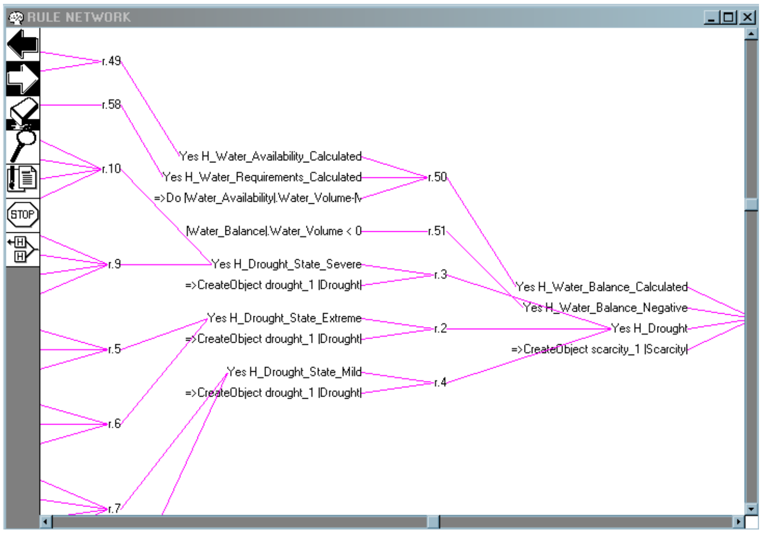


Figure 8: Classification of drought into severe, extreme and mild. In all three cases, a dynamic object drought_1 is created. Also it is declared that in order to compute water balance, both Water Availability and Water Requirements must be investigated.

In figure 9, the monitoring rules of the system are presented. Drought monitoring here is based on hydrologic indices and specifically RDI and SPI. Drought is classified to Severe, Extreme and Mild according to the values of the above indices. Notice that the indices are connected with an OR operator, meaning that only the presence of one is necessary to infer drought.

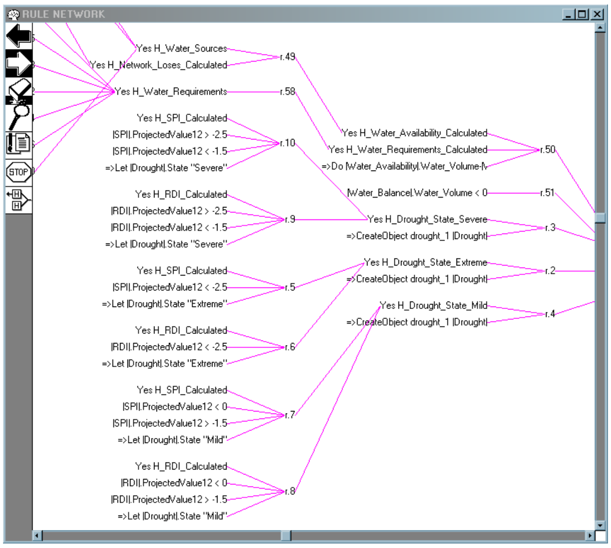


Figure 9: The drought monitoring rules of the system.

In figures 10 and 11 some selected rules are presented and explained, as they were implemented within the NEXPERT OBJECT environment, through the graphical user interface.

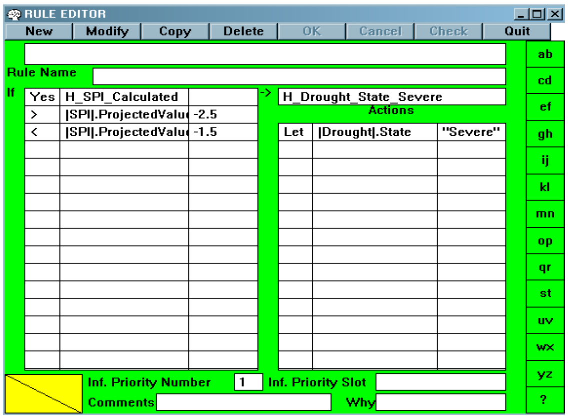


Figure 10: In this rule, the Severe state of drought is explained, using the SPI index. If the value of the index, as projected in 12 month period of time, is more that -2.5 and less than -1.5, then a severe drought state is inferred.

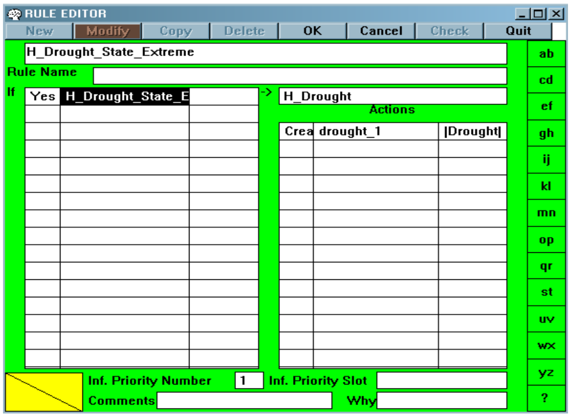


Figure 11: In this rule, the dynamic creation of a drought object is shown, once the drought existence is set to true from the proper hypothesis.

3. Discussion of Results

After the formalization of knowledge, a cyclical development included testing and refining the production rules under the consultancy of the team of experts. Of course an expert system by default is very complex and time consuming software to build. Some simplifications were made and many of the features, that a hydrology expert would like to see implemented, were dropped due to complexity. Nevertheless, the final system consisted of hundreds of rules and is difficult to be presented in every detail.

During the testing period, many branches of the system occurred and lots of revisions were made to the code. Here, we present some important sub-systems of the expert system as one would see them in runtime. This was chosen so that the reader can have an overview of the complexity of the system. Also, figures of the graphical debugger are included, as well as reports that provide the requested explanation facility.

First, the system establishes the parameters of initialization. This is represented through a knowledge island called Initial State. Figures 12 and 13 show some parameters asked from user for the Initial State assessment. During this state of questioning, important parameters of the study area are asked, such as population, network capacity etc. At any point there are reports available to the user to explain the state of the system and the decisions made so far during runtime (Figure 14).

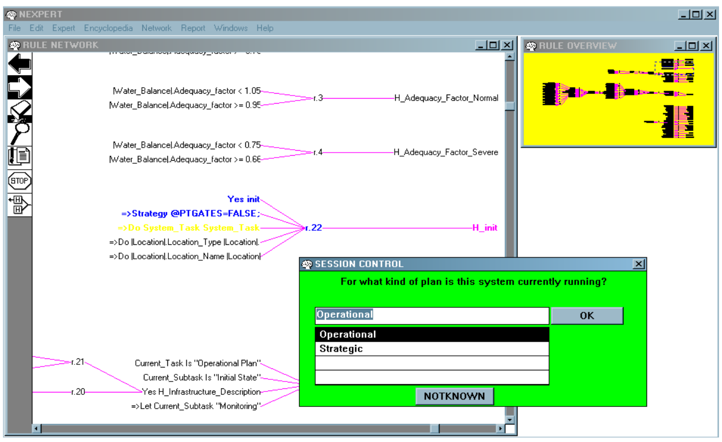


Figure 12: The selection whether to run the system in Operational or Strategic mode.

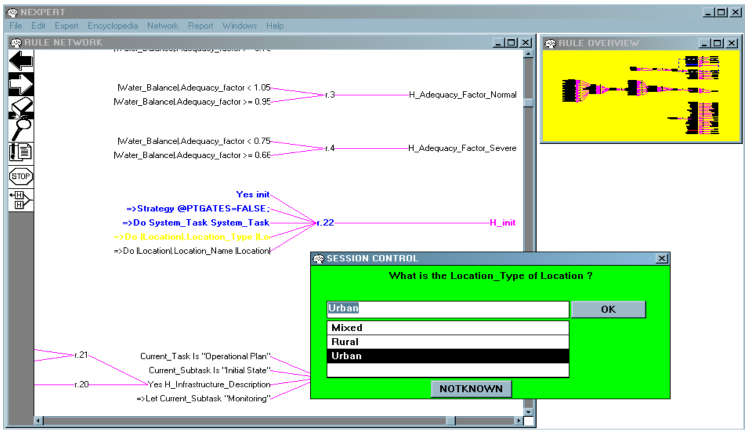


Figure 13: The Location Type is asked to determine the knowledge tree that will become the search domain for inference.

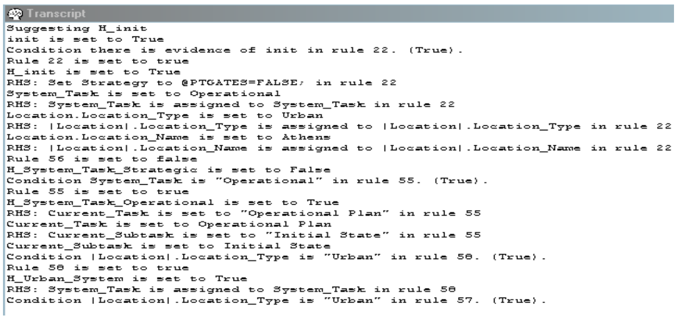


Figure 14: Here, a report window is presented, which is called transcript and records everything that takes place during runtime. The knowledge engineer is able to recall how each decision was made and make proper adjustments or understand better the system behaviour.

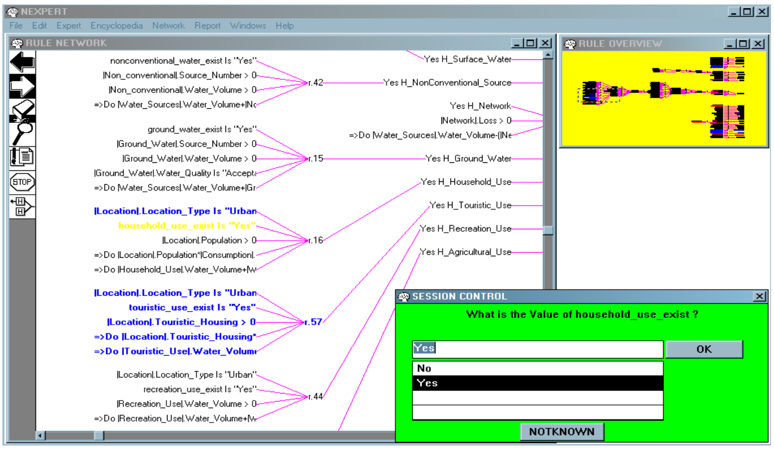


Figure 15: The system asks about household use

Then, the system tries to establish the Water Balance of the area. This is computed from the Water Sources estimation and the Water Usage estimation rules. For example, in Figures 15 and 16 the system asks if there is household use of water in the area and then asks about population, so that to infer by empirical computation the water demand of the area. At the same time (Figure 17) the system is about to ask more questions in order to complete this rule branch (touristic use, household use recreational use etc.). Losses of the network is a very important factor, in order to define the real water requirements, including the water that is lost during the transfer (Figure 18).

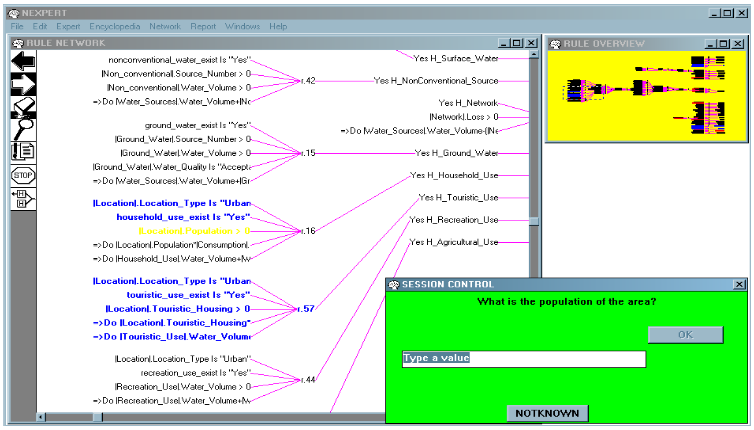


Figure 16: Instead of asking for all the data available in the initialization, such as population of the study area, the system only asks the necessary data to proceed.

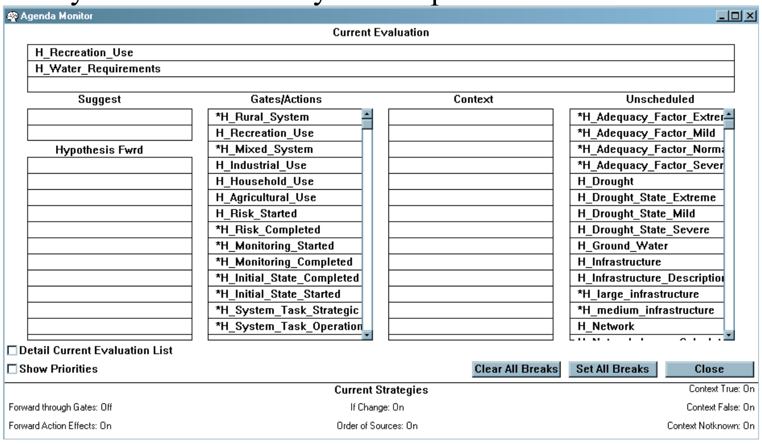


Figure 17: In the agenda monitor, we can observe that for evaluation of Water Requirements, the system moved backward to ask for industrial use, household use and recreational use. At the same time, many other tasks are waiting to be evaluated through the gates mechanism of the knowledge island.

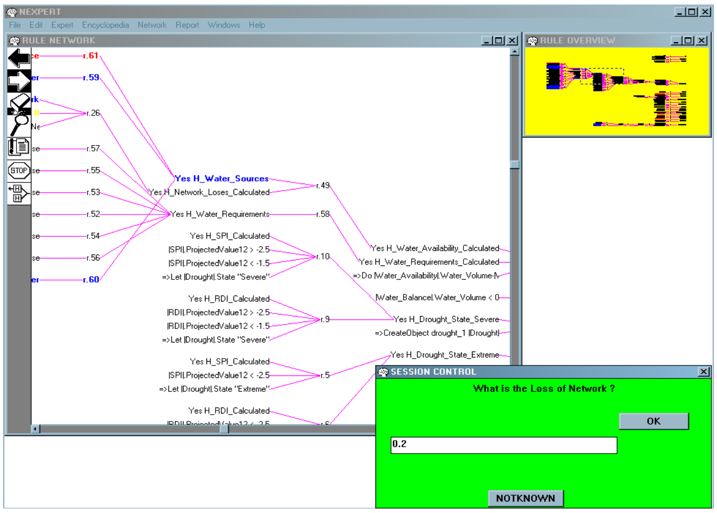


Figure 18: Losses of the network.

The Drought Monitoring stage and the Water Shortage assessment stage follow. At first, drought indices are computed for the study area (Figures 19 and 20). Afterwards, those indices are projected to a 12 month time period. Drought condition is then inferred.

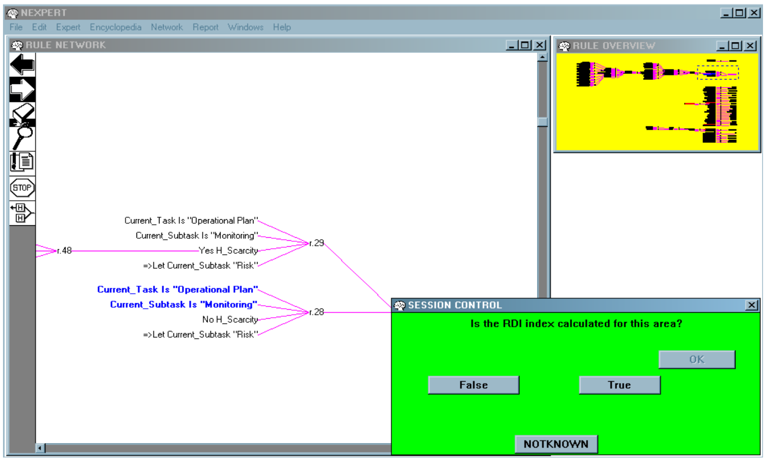


Figure 19: Then, the system starts the monitoring stage by asking the index to be used for drought monitoring purposes.

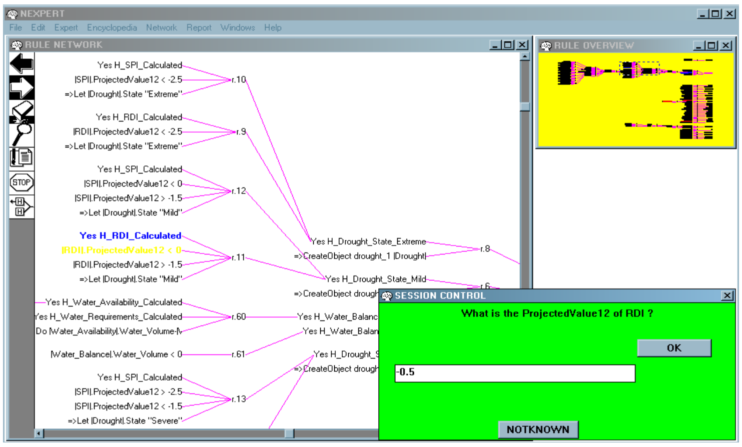


Figure 20: The system asks for the normalized (predicted) value of the index, based on models that are computed externally of the expert system and are provided through user input. If this rule is set to true, drought is inferred and classified into mild, extreme or severe. Other indices can also be used, as they are supported by the system. Also, combinations of indices can be used.

The Water Shortage assessment takes into account all information collected (network status, water sources, water demand, drought indices) and computes the water shortage (if exists). The expert system has the ability to infer drought conditions and depending on the water requirement and water availability data collected, to make an estimation of the water scarcity problem for the area. In later tasks, the system after identification of such conditions, provides suggestions about the measures that need to be taken.

4. Conclusions

The presented formalization in the expert system tool NEXPERT OBJECT was successful and allowed the representation of both complex and simple rule structures that are involved in the water shortage problem. The inferential knowledge was expressed in rules through object oriented structures and AND-OR tree hypotheses. Different strategies were followed, and the path followed by the system, showed that it is very flexible to handle new information and different inference approaches. Also, different paths were followed according to different user input, which is a very strong advantage of expert systems and reduces the search space to the needs of each specific domain problem.

Still, though, many aspects of the problem were not tackled and many simplifications have been made in the “closed world” assumption that many AI paradigms encapsulate. Future efforts will concentrate on better generalization and deeper knowledge acquisition.

5. References

Negnevitsky M., 2005. Artificial Intelligence, a Guide to Intelligent Systems. Pearson Education, p.440, 2005.

Neuron Data, 1991. NEXPERT OBJECT user guide, Neuron Data. Palo Alto, CA.

Neuron Data, 1993. Smart Elements user guide, Neuron Data. Palo Alto, CA.

Argialas D., Miliaresis G., 1996. Physiographic Knowledge Acquisition: Identification, Conceptualization and Representation. Proceedings of the ASPRS, Vol. 3, Baltimore, MD, pp. 311-320.

Argialas D., Miliaresis G., 1997a. Landform Spatial Knowledge Acquisition: Identification, Conceptualization and Representation. Proceedings of the ASPRS, Vol. 3, Seattle, Washington, pp. 733-740.

Argialas D., Miliaresis G., 1997b. An Object Oriented Representation Model for the Landforms of an Arid Climate Intermontane Basin: Case Study of Death Valley/CA. 23rd Conference of the Remote Sensing Society, Reading, U.K , pp. 199-205.

Jackson, P. l986. Introduction to expert systems. - Addison-Wesley, Reading, MA.

Krishnamoorthy C.S., S. Rajeev 1996. Artificial Intelligence and Expert Systems for Engineers. CRC Press LLC ISBN: 0849391253.

Jay Liebowitz, 1997. The Handbook of Applied Expert Systems. CRC Press LLC ISBN:0849331064