Knowledge Representation for a Power Plant Interactive Operating Advisor

by

Gene Maillet

TR90-050, July 1990

This is an unaltered version of the author's
M.Sc.(C.S.) Thesis

Phone: (506) 453-4566
Fax: (506) 453-3566

Abstract

The safe and efficient generation of electricity by power plants requires system operators who are highly competent and knowledgeable in their field. The primary source of knowledge for these system operators is the power plant system operating manuals which are prepared for all power plant systems which need to be operated and are fully comprehensive. The following document describes a generic expert system capable of incorporating the knowledge contained in these manuals. The approach to building the expert system is unique. Instead of using rules and frames directly to represent the operating manual knowledge, building blocks of different types of reasoning, called generic tasks, have been identified. Each building block is generic and reusable as a component of a more complex reasoning task. The expert system is able to advise the system operator on the appropriate course of action during system start-up, running, and shutdown, under both normal and abnormal operating conditions. A prototype expert system entitled "Power Plant Interactive Operating Advisor" was built for the Point Lepreau Liquid Zone Control System. A discussion is given on automating the knowledge acquisition process for power plant operating manuals.

Table of Contents

Acknowledgements

This thesis was completed over a two year period, mostly on a part-time basis. I would like to acknowledge the support given to me by my employer, NB Power, and particularly the management group of the Information Systems Division. Without their help, this thesis would not have been possible.

I would also like to thank Professor Brad Nickerson, my thesis advisor, for his patience and supervision. His timely guidance kept me moving forward when there were many paths to choose from.

The staff at Point Lepreau Generating Station, particularly Bryan Patterson, served as the operating manual problem domain expert. I wish to express my sincere appreciation to Bryan for his time and expertise.

Finally, I wish to thank my wife, Jacqueline, and children, Amy and Stephanie, for their support and encouragement.

List of Figures

Chapter 1 INTRODUCTION

1.1 Problem Definition

The New Brunswick Electric Power Corporation is a publicly owned Provincial Crown Corporation established in 1920. Its mandate is to safely and efficiently generate and distribute electricity to all areas of the province. This thesis is the result of their strong commitment to the safe operation of their power plants. The primary goal consists of developing a generic knowledge representation structure capable of encapsulating the knowledge contained in power plant operating manuals. Several prototype expert systems were built.

The generation of electricity takes place at power plants of many types located throughout the province. The safe and efficient operation of these power plants is the responsibility of the power plant system operators. The power plant operating environment consists of an industrial grade computer which monitors the various plant systems and annunciates any deviation to the operator via an alarm or display. The system operator must then correct the situation relying upon his knowledge, expertise, and the system operating manual. The positive outcome of a normal or abnormal annunciation depends upon the system operators following the detailed approved operational procedures elaborated upon in the power plant system operating manuals.

The current power plant operating environment within which the system operators must function consists of a control room with the necessary computer equipment to monitor and control the power plant systems and a carousel containing approximately two hundred system operating manuals totalling ten thousand pages of technical information. These operating manuals represent the primary source of knowledge for the system operators and it is critical that the information contained be readily accessible. Time is often limited and it is important that the operator have immediate access to the appropriate section of the appropriate operating manual. There is a problem in accessing the appropriate manuals and the different sections of the same manual quickly enough, especially when alarms require immediate attention.

This thesis is an attempt to improve the availability and use of the huge knowledge-base embodied by the power plant operating manuals. An expert system approach will be used to capture this knowledge and make it available to the system operators. The resulting expert system is at the disposal of the power plant system operators during all stages of plant operation. The operator may consult with the expert system which will analyze the data given to it from the control computer and recommend an appropriate course of action. The expert system uses the knowledge encapsulated from the appropriate power plant system operating manual to recommend an appropriate course of action to follow. The expert system can quickly identify the sections of the system operating manual which are relevant in the current situation and discard pages of information which are not applicable. The expert system is also capable of walking the system operator through the approved operational procedures, ensuring that each step is correctly performed.

The main emphasis of this thesis is on developing a generic knowledge representation structure capable of encapsulating the knowledge contained in the power plant operating manuals. Various researchers believe that expert system problem domains can be classified into one of several generic categories [Hayes-Roth, Waterman, and Lenat 1983]. The power plant operating manual problem domain does not fit nicely into one of these categories and a novel knowledge representation structure is required. The interface to the power plant control computer, which can ultimately provide plant status and parameters to the expert system, is simulated by manually setting the initial states of all objects referred to by the Liquid Zone Control System.

1.2 Problem Domain (Operating Manuals)

Power plant operating manuals represent the accumulated wisdom of a large number of people devoted to the safe and efficient generation of electricity. Generally, a separate operating manual will exist for each system within a plant. Operating manuals are prepared for all plant systems which need to be operated and are fully comprehensive [Johnson 1988]. NB Power has published a document entitled "How to Write a Power Plant Operating Manual, RD-01364-P2" which describes in detail the structure and content of power plant operating manuals [Johnson 1988]. In addition, Point Lepreau staff has taken measures to ensure that a comprehensive set of acceptable operating manuals are produced and that administrative procedures are established to ensure proper control and revision of these manuals.

The correct and safe operation of a power plant hinges upon the system operators following the detailed approved operational procedures elaborated upon in the power plant operating manuals. More specifically, plant operating manuals contain [Johnson 1988]

1. Normal operations for start-up, running and shutdown.
2. Abnormal events which are foreseeable and appropriate operator action.

3. Additional information to permit system monitoring, trouble shooting and, where appropriate, corrective action.

All power plant operating manuals adhere to a standard format with similar information in sections with standard headings. Figure 1-1 illustrates the recommended section headings by showing the table of contents for a typical power plant operating manual. Duplication of information within an operating manual is avoided by precisely defining the content of each of the 11 sections.

Section 1 (System Scope) gives a short description of the system covered by the operating manual. The purpose of this section is to ensure there is no confusion as to which system is covered by this manual [Johnson 1988].

Section 2 (Operational Flowsheet) contains references to the applicable system flowsheets, location diagrams, and control schematics.

Section 3 (Operating Rules and Limits) is divided into two groups. The first group (Licensing Requirements) contains operating rules and limits which guarantee compliance with operating licenses and are repeated verbatim from the original document. The second group (Other Requirements) consist of rules and limits which may relate to NB Power

ENERGIE NB POWER

POINT LEPREAU G.S.

NUCLEAR OPERATIONS

FUNCTION	SYSTEM	UNIT	BSI
OPERATING MANUAL	STORAGE TRANSFER & RECOVERY	1	33330

0.0 INDEX

0.0 INDEX

1.0 SYSTEM SCOPE

2.0 OPERATIONAL FLOWSHEET

3.0 OPERATING RULES AND LIMITS

3.1 Licensing Requirements
3.2 Other Requirements

4.0 SYSTEM HAZARDS

4.1 Equipment
4.2 Personnel

5.0 NORMAL OPERATION

5.1 Shutdown State
5.2 Start up Procedure
5.3 Operating State
5.4 Shutdown Procedure
5.5 Other Routine Operations

6.0 ABNORMAL OPERATION

7.0 ACTION FOLLOWING TRIPS AND ALARMS

7.1 Trips
7.2 Alarms

8.0 FAILURE OF AUXILIARY SERVICES

8.1 Electrical
8.2 Air
8.3 Water
8.4 Computers
8.5 Others

9.0 CHEMICAL CONTROL

| 10.0 TEST INDEX

11.0 REFERENCES

| APPENDIX 1 VALVE AND HANDSWITCH LIST

APPENDIX 2 INTERLOCK DIAGRAMS

| APPENDIX 3 S.O.S. INDEX

RD-01364-P2
Rev. 4
Appendix 6
Page 1 of 1

PREPARED BY	APPROVED BY	DATE	REV	PAGE
System Engineer	Technical Sup.		0	1 OF xx

Figure 1-1 Recommended Section Headings for a Power Plant Operating Manual.

practice, personnel safety, economic considerations, etc. [Johnson 1988].

Section 4 (System Hazards) contains a brief list of the primary personnel and equipment hazards associated with the system [Johnson 1988]. Unexpected or unusual hazards are emphasized.

Section 5 (Normal Operation) include shutdown state, start up procedure, normal operating state, and shutdown procedure. Procedures other than shutdown and start up which are routine are also included in this section.

Section 6 (Abnormal Operation) describes corrective procedures which cover situations which are foreseeable but not considered normal and therefore not covered in the normal operation section.

Section 7 (Action Following Trips and Alarms) contain both automatic action and recommended operator action following a computer annunciated alarm. All possible system alarms are listed and categorized by priority.

Section 8 (Failure of Auxiliary Services) describes equipment response to a failure of one of the following auxiliary services: electrical, instrument air, water, computers, and other. Required operator action to a failure is not covered in this section, but rather in the appropriate section such as abnormal operations, action following trips and alarms, etc. [Johnson 1988].

Section 9 (Chemical Control) identifies the sections of the system which require chemical control.

Section 10 (Test Index) contains only an index to the relevant system tests [Johnson 1988]. The procedures for all regularly performed tests are in the operational testing manual.

Section 11 (References) gives a list of relevant reference material.

The current thesis utilized the "Liquid Zone Control System" operating manual as the domain of knowledge for the prototype expert system. The Liquid Zone Control System is one of several reactivity control devices in the reactor [Parker and Leclair 1987]. The system consists of a closed circuit demineralized water system and a closed circuit helium cover gas system. Reactor power is controlled by varying the amount of light water contained in the fourteen zone compartments which penetrate the reactor from above [Parker 1980]. Figure 1-2, taken from the Liquid Zone

Control Training Manual [Parker 1980], illustrates the architecture of the Liquid Zone Control System. The Liquid Zone Control System operating manual was proposed by Point Lepreau staff as being representative of power plant operating manuals in general. Pertinent sections of this operating manual were judiciously chosen in order to get a complete representation of the types of knowledge contained in the power plant operating manuals. Collaboration with Point Lepreau power plant staff provided a second source of expertise.

1.3 Knowledge Representation and Expert Systems

Expert system development began in the 1960's and the first systems began appearing in the early 1970's. In recent years expert systems or knowledge-based systems have had many important successes. Among the most renowned are Dendral [Lindsay et al. 1980], MacSyma [Martin and Fateman 1971], Mycin [Shortliffe 1976], Hearsay II [Erman et al. 1980], and R1 [McDermott 1980].

Emerging as a practical application of artificial intelligence, they are fast becoming accepted as solutions to real world problems which defy conventional programming methods [Hayes-Roth, Waterman, and Lenat 1983]. Individuals responsible for operating complex equipment such as nuclear

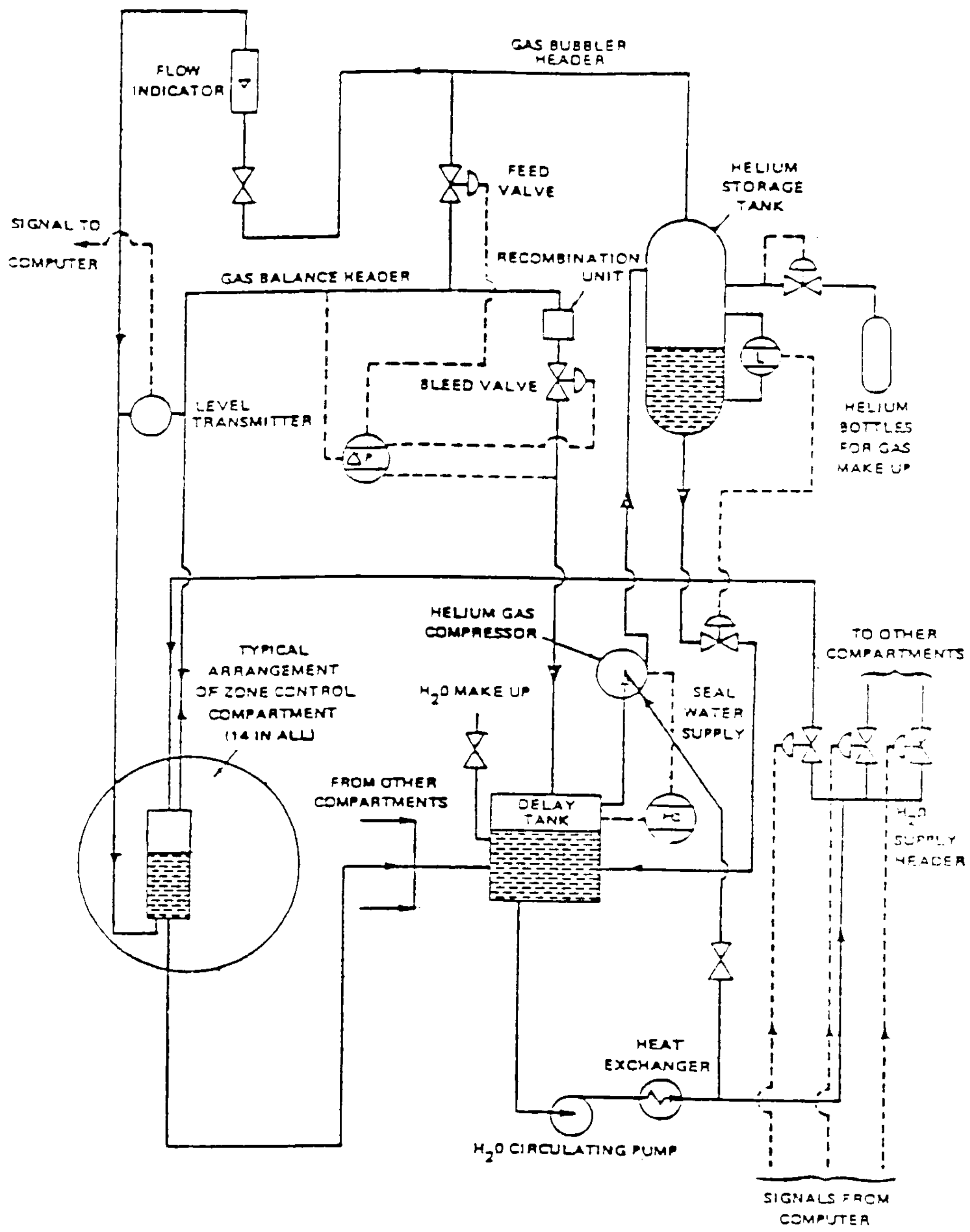


Figure 1-2 Liquid Zone Control System Architecture

reactors will soon be assisted by expert systems that monitor the equipment, anticipate problems, and make intelligent suggestions to the operators [Harmon and King 1985]. Training will also be revolutionized by the introduction of expert systems.

Expert systems have been defined as computer-based systems that use knowledge, facts, and reasoning techniques to solve problems that normally require the abilities of human experts [Martin and Oxman 1988]. Expert systems are geared towards applications which are knowledge intensive. The power of such a system lies in its ability to encapsulate the knowledge about a problem domain, thus enabling it to replicate the reasoning of a human expert.

Expert systems have been designed and implemented in many different ways. In general, a working expert system has two major components: a knowledge-base containing expert-level knowledge on a particular problem domain and an inference engine for processing this knowledge. Other components such as a knowledge acquisition facility, an explanation facility, and a user interface are also commonly found in expert systems. Figure 1-3 illustrates the architecture for a simple expert system.

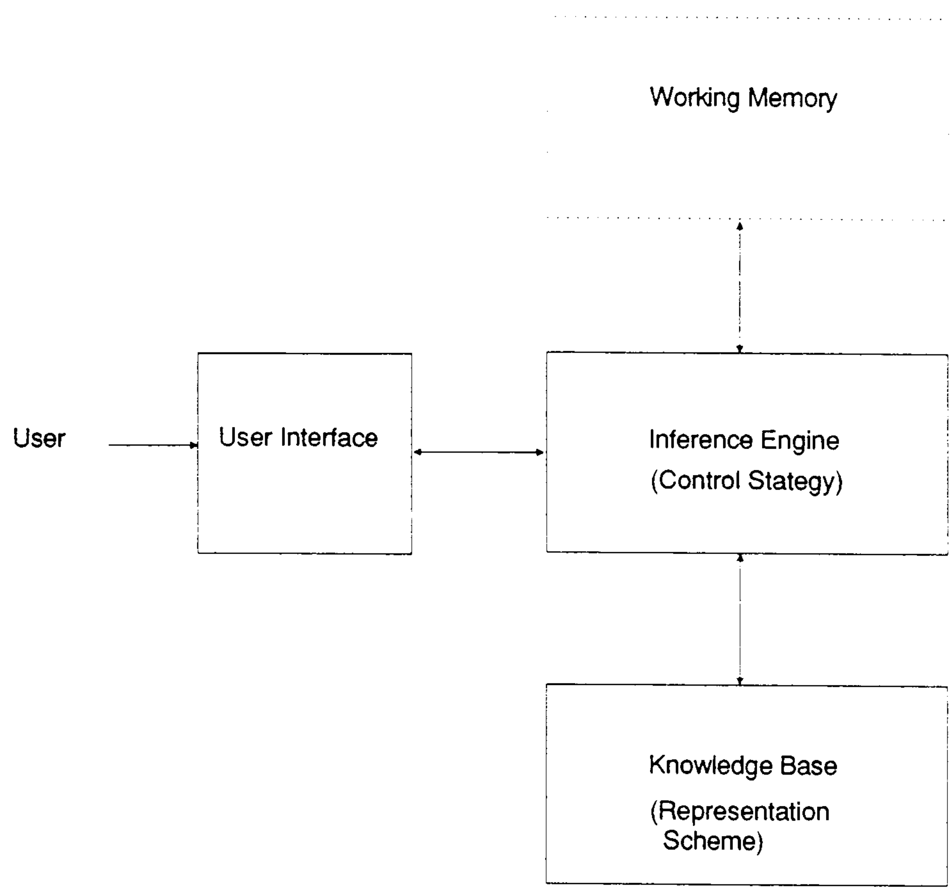


Figure 1-3 Expert System Architecture

It is agreed that the most difficult task in building an expert system is the identifying, extracting, and reformulation of the expert knowledge concerning a specific problem domain [Hayes-Roth, Waterman, and Lenat 1983]. Many different representation schemes exist for capturing the expert's knowledge in a form recognizable by the computer. The most common knowledge representation schemes are rules, frames, and semantic networks [Gartner Group 1989].

Expert systems using the rule representation scheme are often called production systems. A rule is composed of two parts: an antecedent condition (premise) and a consequent proposition (conclusion). The antecedent and consequent of a rule represent condition-action pairs: IF condition THEN action. Rules can be simple or very complex. Two possible enhancements to the rule representation scheme are certainty factors and pattern-matching variables. Attaching certainty factors to the antecedent and consequent of rules allow rules to generate conclusions with less than 100% certainty. The use of variables provide for the assimilation of many rules into one rule which is more general.

Semantic networks are one of the oldest and most powerful representation schemes in AI. A semantic network is a collection of objects called nodes connected by arcs or links. Generally, both the nodes and links are labelled. The nodes can be physical objects, conceptual entities such as acts and events, or descriptors such as large and yellow. Links define a relationship between the two objects it connects. There is no restriction on what type of relationship a link may represent. Some of the more common relationships are "IS-A", "HAS-A", and "PART-OF". Some of the benefits of the semantic network representation scheme are it's flexibility and ability to support inheritance. Inheritance refers to the ability of a child node to inherit the characteristics of it's parent node. Figure 1-4,

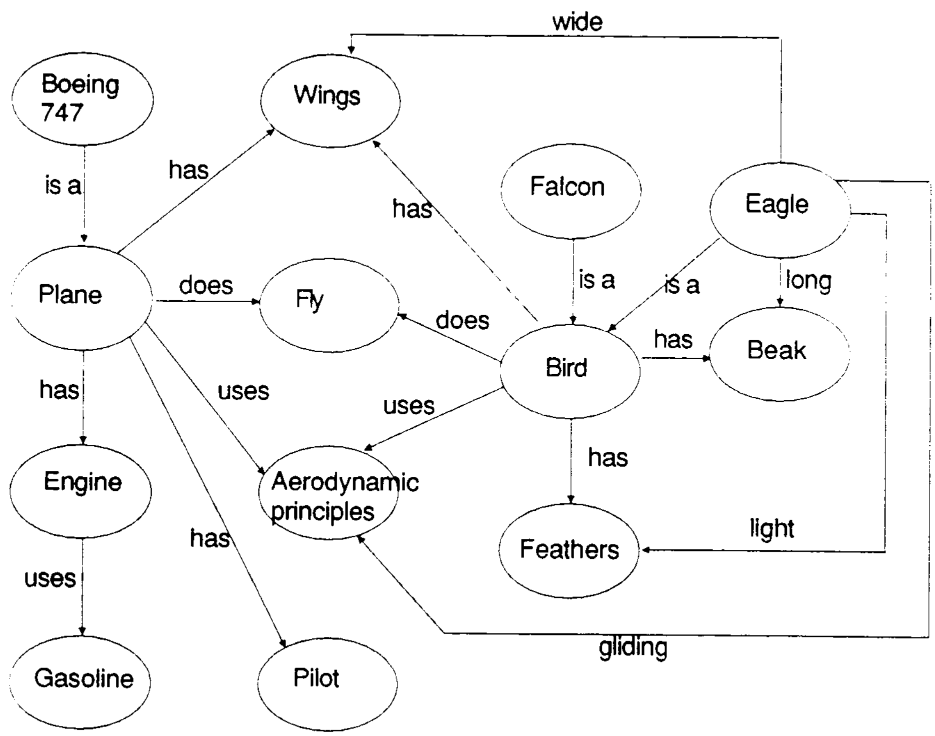


Figure 1-4 Semantic Network Showing Relationships Between Parts of a Bird and a Plane.

reproduced from [Levine, Drang, and Edelson 1986], illustrates a semantic network showing relationships between parts of a bird and a plane.

Frames, originally developed by Minsky [Minsky 1975], provide another popular mechanism for representing knowledge. A frame is a named collection of slots that describe a class of objects. Filling in the slots of a frame with specific slot-values creates what is commonly known as an instance of that frame. Slots may be assigned default values, a valid type, a valid range, pointers to other frames, or procedures to be executed when the slot-value is modified. Frames may be linked together into a hierarchy to allow for inheritance which provides for a powerful semantic net representation of the knowledge-base. Figure 1-5 illustrates a sample frame representing a computer and a specific instance of the frame computer.

The second major component of an expert system is the inference engine. The inference engine is the component that performs the inference reasoning for the expert system. It is responsible for deciding how and in what order the knowledge contained the knowledge-base will be used [Martin and Oxman 1988]. It uses the expert-level knowledge represented in the chosen knowledge representation scheme to infer new knowledge about the problem domain. Although the inference engine plays an important role in the expert system architecture, it is relatively simple and the power of the expert system is attributed to the expert-level knowledge contained in the knowledge-base.

Frame for Class Computer

Make:	
Processor:	
Speed:	
Diskette 1:	1.2 M or 1.4 M
Diskette 2:	1.2 M or 1.4 M
Video:	
Base Memory:	Default=640 K
Extended Memory:	
Hard Drive:	

Specific Instance of
the Frame Computer

Make:	IBM
Processor:	Intel 286
Speed:	10 Mhz
Diskette 1:	1.2 M
Diskette 2:	1.4 M
Video:	VGA
Base Memory:	640 K
Extended Memory:	384 K
Hard Drive:	40 M

Figure 1-5 Sample Frame Representing the Class Computer and a Specific Instance of the Frame Computer.

Two of the most popular inference mechanisms are forward chaining and backward chaining. Forward chaining and backward chaining are control strategies used to specify how the rules contained in the knowledge-base are to be used and in what order they are to be executed.

Systems using the forward chaining inferencing mechanism are often called data driven systems. This reasoning mechanism attempts to reason from facts to conclusions. The antecedent conditions of all the rules are examined to determine which are satisfied given the set of current known facts. One rule is selected for firing based on a rule conflict resolution algorithm. The conclusions generated by the rule consequent are added to the set of known facts and the system examines the rules again. Most experts agree that the forward chaining inference mechanism of rules is typically best suited for tasks that respond to a stream of input data, such as monitoring problems [Grimson and Patil 1987]. One of the best algorithms for implementing forward chaining is the Rete algorithm developed for the OPS expert system language [Forgy 1979]. This algorithm reduces the amount of effort the expert system must expend on matching rule antecedents to facts to a bare minimum.

Backward chaining is also known as goal-directed chaining. This reasoning mechanism first establishes a set of tentative working hypotheses. One hypothesis is selected and called the goal. The expert system then uses the rules to work backwards to prove the goal true. The consequents of rules that conclude this goal are examined to determine if their antecedent conditions are satisfied. If not, these antecedents conditions may be established as sub-goals and the process is recursively re-applied. Backchaining continues until the validity of the sub-goals are known and conclusions can be drawn about the original goal. If all the possible hypotheses (goals) are known and they are small in number, then backward chaining is very efficient [Harmon and King 1985].

Another characteristic which distinguishes inference engines is whether they support monotonic or nonmonotonic reasoning. In a monotonic reasoning system, the total number of facts in the knowledge-base is always increasing. Once a fact becomes true, it remains true for the entire consultation and the information in the knowledge-base increases monotonically. In a nonmonotonic reasoning system, facts can be both asserted and retracted from the knowledge-base as more information may prove an earlier decision incorrect. The total number of facts in the knowledge-base is not constantly increasing. Retracting a single fact from the knowledge-base is not difficult, but tracking down all the implications that are based on this fact can be extremely complex. Most expert systems support monotonic reasoning or a restricted version of the nonmonotonic reasoning system [Harmon and King 1985].

Chapter 2 ANALYSIS OF DIFFERENT APPROACHES

Various solutions to providing quick and easy retrieval of the massive amount of complex information contained within the power plant operating manuals are being considered by NB Power. Handling this huge knowledge-base seems to defy conventional programming techniques and an innovative approach is required to make this knowledge quickly accessible to the system operators when time is limited [Patterson 1989].

Third generation languages consist primarily of algorithms and data which do not provide a manageable way to represent complex knowledge in a natural form. Algorithms, which can be represented by program flowcharts, determine how to solve a specific problem. The data represents the parameters of the specific problem at hand. The flow of execution within the program is pre-determined and tightly structured. In the power plant operating manual problem, actions need to be taken as soon as certain conditions exist in the power plant system. Using a third generation language, it would be extremely difficult to be sure that all statements are in the right location so that they are executed at the right time. Every procedure of an operating manual could potentially result in a new algorithm being written. It has been recognized in the literature that human knowledge does not fit the traditional model consisting of algorithms and data [Hayes-Roth 1984].

The use of hypertext is another approach to the operating manual problem. Although hypertext may prove to be a useful tool in accessing the operating manual knowledge, it cannot interpret this knowledge and therefore cannot advise the system operator on the appropriate course of action and justify its reasoning. A very powerful knowledge representation scheme and the ability to reason about objects and their relationships to one another is necessary to encapsulate the operating manual knowledge and make logical decisions based on this knowledge.

An approach that seems promising is the application of expert system technology to the power plant operating manual problem. It is often said that the most important factor to consider when building an expert system is choosing the right problem. Power plant operating manuals lend themselves well to expert system technology for numerous reasons as outlined below.

The knowledge about each power plant system has been precisely defined in the power plant operating manuals. Therefore, the identification and extraction of the problem domain knowledge required in the expert system is greatly simplified. Many experts agree that in fields where the knowledge can be precisely defined expert systems will come into their own very quickly [Clayton 1987].

The primary task that remains is determining an appropriate knowledge representation scheme for the power plant system operating manual. An expert system knowledge-base allows for a natural representation of the power plant operating manual knowledge, resulting in a knowledge-base which is easily understood and modified. Through the use of facts and frames the knowledge engineer can structure the knowledge-base to represent almost anything [Clayton 1987]. The power plant system operating manuals would be well represented by frames with dynamic inheritance.

A characteristic which distinguishes expert systems from procedural software systems is the order in which code gets executed. In an expert system, the order in which rules appear is not important, as the rules will be fired opportunistically, as the situation demands. An expert system applies rules when the appropriate conditions are met, regardless of the location of the rules in the program. This is commonly known as the data driven approach. Some forward chaining expert system shells implement this data driven approach well. The eventual link up of the expert system with the power plant control computer makes the data driven approach imperative.

The operational procedures found in the power plant operating manuals often require several tasks be performed at the same time. This type of program execution is not easily accomplished using systems developed with traditional programming languages. With a data driven expert system, this is possible because at each cycle the inference engine examines all rules to determine which rules are fireable. The rule with the highest priority from the set of rules that are fireable is then fired and another cycle begins.

Another characteristic of expert systems is that they provide a separation of knowledge from control. Control is the part of the expert system that does the processing, that is deciding what and when things get done. Knowledge is what the human expert knows about the problem domain. The separation of knowledge from control provides an environment where the control or knowledge can be extended or enhanced independently of the other.

An expert system does not suffer from fatigue, tunnel vision, or other human ailments. Some experts believe that the accidents at Three Mile Island and Chenobyl nuclear power plants can be directly attributed to operator error.

An expert system would be able to advise the system operator on the appropriate course of action during system start-up, running, and shutdown, under both normal and abnormal operating conditions.

In addition to recommending an appropriate course of action, the expert system would, if necessary, be capable of explaining its reasoning to the system operator. The explanation facility examines the knowledge-base to determine how or why it arrived at the particular conclusion [Grimson and Patil 1987]. The system operator needs to trust the recommendations given by the expert system. By offering to explain how it reached its conclusion, the system conveys an impression of reasonableness [Hayes-Roth 1984].

These features have the effect of improving power plant safety and provide an excellent training tool for novice system operators. An expert system can provide for better training of individuals responsible for operating the complex equipment found in power plants. Some experts believe that the introduction of expert systems will revolutionize training [Harmon and King 1985].

The application of artificial intelligence, and more specifically expert systems, to the power industry is a very active and promising research area [EPRI NP-4141; EPRI EL-4323]. For example, Atomic Energy of Canada Limited (AECL) has a development program underway which investigates the use expert systems to assist operations staff of future CANDU plants [Lupton, Anderson, and Basso 1989]. The Point Lepreau Generating Station is taking an aggressive approach to improving their Emergency Operating Procedures (EOP's). Work is underway to automate the EOP's and ultimately replace the paper manuals with the computerized system [Storey, Johnson, Patterson, and Stafford 1988]. In addition to improving the technical content of the EOP's, Point Lepreau is also researching potential improvements in the man-machine interface [Johnson, Patterson, and Pilkington 1988].

Another factor which makes an expert system approach possible is the commercial availability of several high-end expert system building tools or shells. These tools facilitate the building of an expert system and are the result of many years of research in the AI field. The use of an expert system shell permits the development of an expert system in an order of magnitude less time than is possible with traditional artificial intelligence programming languages [Gevartar 1987].

The application of expert system technology to the power plant operating manual problem seems to be the most promising solution and it warrants further research. The potential benefits of incorporating the power plant operating manual knowledge into an expert system are numerous.

Chapter 3 A GENERIC SOLUTION

A generic solution is required so that the knowledge from any power plant system operating manual can be represented in the proposed expert system without rethinking the entire solution. There exists approximately 200 power plant system operating manuals at the Point Lepreau Nuclear Generating Station alone. An expert system built for one of these manuals should be able to be used for the remaining manuals by supplying the knowledge contained in these manuals to the expert system. Rethinking the entire solution should not be necessary.

As previously stated, an expert system approach provides the separation of knowledge from control. The separation of knowledge from control is extremely important in the power plant system operating manual problem because this allows a generic solution to be built. Separating knowledge from control is similar to a human power plant system operator having expertise in two areas. The first area would be control, that is how power plant system operating manuals are structured and how to interpret and use the knowledge found within them. The second area of expertise is the operating manual specific knowledge. The generic part is the first part; that is, it is the same for all power plant system operating manuals. This information is primarily represented using generic frame templates and rule groups. The second part is declarative information. This is represented using a semantic network of frames based on the frame templates. Representing any power plant system operating manual in the proposed expert system is done by supplying the second part only.

Thus the proposed expert system can be thought of as containing two major components: the first component being generic and the second being specific. The generic component forms the expert system structure and control. The specific component consists of the operating manual declarative knowledge which gets processed by the generic component. See Figure 3-1.

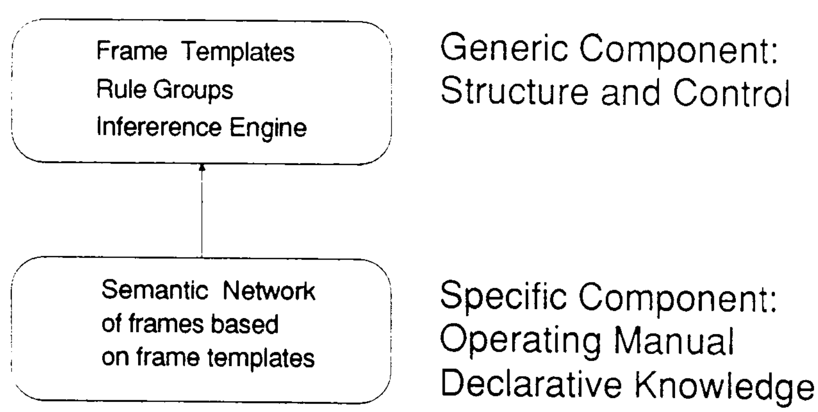


Figure 3-1 Interactive Operating Advisor Generic and Specific Components

The generic component of the proposed expert system is comprised of generic frame templates and rule groups. Frame templates were introduced into the expert system to facilitate the knowledge acquisition process. Frame templates are used as patterns or shells which are copied and filled in to describe operating manual declarative knowledge. Assigning values to the slots of a frame is commonly called creating an instance of the frame. All the instances of the frame templates compose the specific component of the expert system.

There are two types of frame templates: operating manual frame templates and equipment frame templates. Operating manual frame templates are used to describe the operational procedures and other information contained in the operating manuals. Equipment frame templates are used to describe the physical system being discussed in the operating manual. Operating manual frame templates are described in Section 3.2 and equipment frame templates in Section 3.3.

A generic rule group is simply a group of rules developed to perform one task. The rule groups may be thought of as sitting there waiting for the right conditions to exist in the system and then coming alive to perform the task. Rule groups will be explained in detail in Section 3.4. Figure

3-2 illustrates the complete Interactive Operating Advisor Architecture.

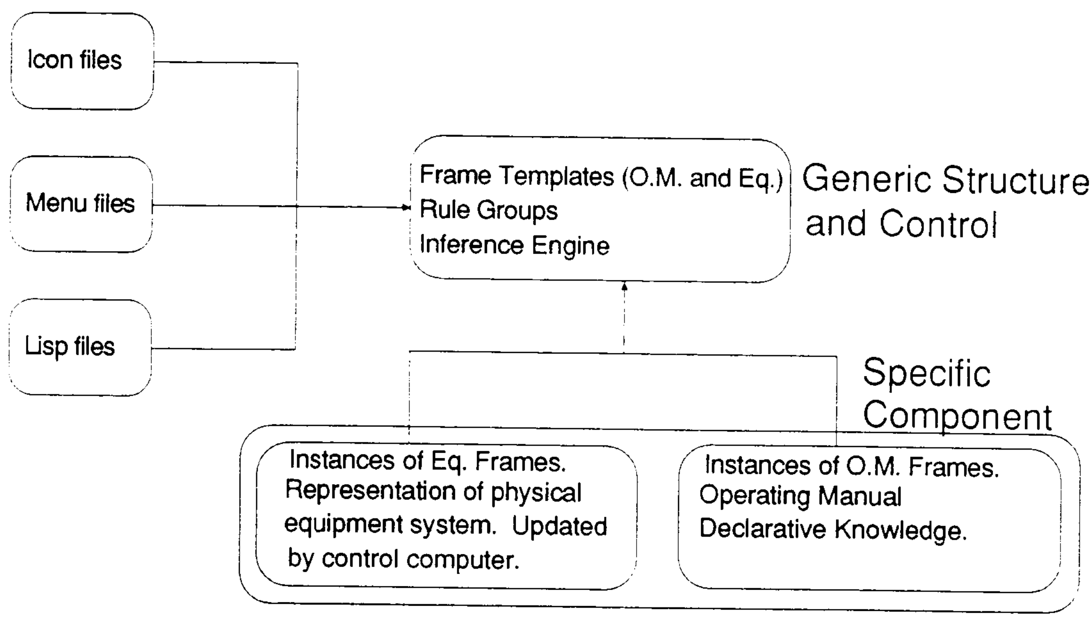


Figure 3-2 Interactive Operating Advisor Architecture

3.1 Generic Tasks

The generic component which forms the structure and control of the expert system has been developed to process the declarative knowledge found in power plant system operating manuals. This declarative knowledge comes in the form of detailed approved operational procedures consisting of one or more steps (see Section 1.2).

Studying several power plant system operating manuals [Dean 1986; Parker and Leclair 1987] resulted in the important discovery that the steps of the operational procedures often require the same type of operator action, but on different equipment items. A result of this finding was that the operating manual knowledge can be broken down into small elements of work which require similar processing. These elements have been called generic tasks.

Each procedure step may require one or more of these generic tasks to be performed. Generally, each procedure step requires a generic task to display to the system operator the procedure step description verbatim from the operating manual for acknowledgement, a generic task to perform one or several actions, a generic task to give a completion message, and a generic task to link to the next procedure step. The generic tasks used to represent a procedure step are processed sequentially.

For each generic task that has been identified one operating manual frame template and one rule group will exist. The operating manual frame template provides a means for representing the task in a form understandable to the expert system. The rule group is responsible for overseeing the execution of this task. Figure 3-3 illustrates the relationship between operating manual declarative knowledge, generic tasks, operating manual frame templates, and rule

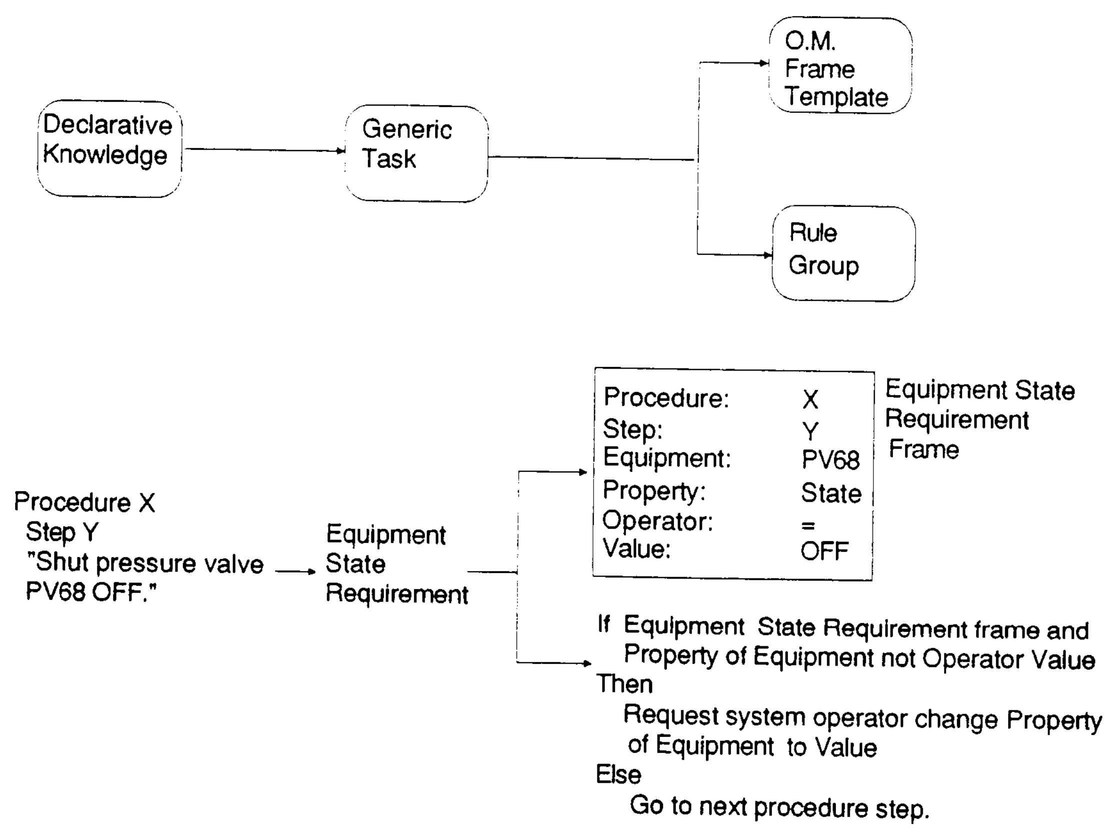


Figure 3-3 Summary of Relationships

groups. It also shows an example of equipment-state-requirement generic task.

An analysis of several power plant system operating manuals [Dean 1986; Parker and Leclair 1987] and Point Lepreau's manual on "How to write a power plant operating manual" [Johnson 1988] has identified the primary tasks for which generic operating manual frame templates and rule groups must be developed. Because of Point Lepreau's standardization of the power plant system operating manuals, the number of tasks is limited. The operating manual frame templates and rule groups for these tasks can process the knowledge contained in any power plant system operating manual.

The primary tasks for which operating manual frame templates and rule groups have been developed are as follows:

1. <u>Equipment-state-requirement-with-and-logic.</u>

This requirement specifies that a list of equipment items be in a particular state. Each entry in the list requires specifying the target equipment and the desired state. Each target equipment may be a single equipment item or several equipment items separated by the addition (+) or subtraction (-) arithmetic operators. Examples of target equipment are pressure valve 68 and tank1 + tank2. The desired state for each entry in the list consists of which slot of the equipment item frame to use, which logical operator to use (<, <= , > , >=, =, <>), and what value to use. An example of desired state is pressure >= 1150 kpa. All the target equipment in the list must be in the specified state to complete this task.

Figure 3-4 illustrates this generic task by showing a procedure step requiring 3 valves be in a specific state.

> Step 1. **Close valves V 73, V 78, and V 103 to isolate the balance header.**
>
> Figure 3-4 Procedure 5.2.1, step 1, from page 8 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating equipment-state-requirement-with-and-logic.

2. Equipment-state-requirement-with-or-logic.

This requirement specifies that one of a list of equipment items be in a particular state. Each entry in the list requires specifying the target equipment and the desired state. Each target equipment may be a single equipment item or several equipment items separated by the addition (+) or subtraction (-) arithmetic operators. The desired state for each entry in the list consists of which slot of the equipment item frame to use, which logical operator to use

(<, <= , > , >=, =, <>), and what value to use. This task is complete as soon as one of the target equipment in the list is in the specified state.

This generic task is illustrated in Figure 3-5 which shows a procedure step requiring that the reactor regulatory system be on in either DCCX or DCCY mode.

> Step 1. **RRS must be ON in DCCX or DCCY. Enter 14 zone level nominal setpoints of 50% (KB 2, PB 27).**
>
> Figure 3-5 Procedure 5.2.7, step 1, page 11 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating equipment-state-requirement-with-or-logic.

3. Select-equipment-item.

Selection of an equipment item from a list for future processing. Many of the power plant systems have several equipment items performing the same function. Upon startup, one is selected for use and the others serve as backups. This generic task presents the options to the system operator for selection and remembers which equipment item was selected.

Figure 3-6 illustrates this generic task by showing a procedure step requesting the system operator select one

Step 1. Select one pump to OFF.

Figure 3-6 Procedure 5.2.4, step 1, from page 10 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating select-equipment-item.

pump and change it's state to off.

4. <u>Monitor-equipment-item.</u>

This generic task consists of monitoring the state of an equipment item. Any deviation from normal is immediately signalled the system operator for resolution. It is very common that one or several equipment items must be carefully watched during the execution of part or all of an operational procedure. For each equipment item it is possible to specify which slot of the equipment item frame to use, which logical operator to use (<, <= , > , >=, =, <>), and what value to use. This generic task has associated with it a stop monitoring procedure step. If any of the equipment items deviate from normal, the deviation is brought immediately to the system operator's attention.

Figure 3-7 illustrates this generic task by showing an operator action requiring the monitoring of reactor power and reactor power trend following a window alarm indicating

2. Check and monitor
reactor power to ensure
that it has stabilized
by the action of RRS
using the increased
opening of zone level
control valves and/or
absorber/adjuster
action. Monitor reactor
power display unit
(panel PL 6), Lin N
chart recorder (panel
PL 4), and reactor power
trend (KB 1, PB 49).

Figure 3-7 Window Alarm CI 0432, operator action 2, from page 27 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating monitor-equipment-item.

helium balance header pressure abnormal.

5. <u>Find-equipment-item.</u>

Find an equipment item with a specified property in a specified state. The expert system remembers what equipment item was found and makes it available for future processing.

This generic task is illustrated in Figure 3-8 by showing a procedure step requesting the system operator switch the running pump to off. The expert system must determine which pump or pumps are running.

Step 2. Select running pump to OFF.

Figure 3-8 Procedure 5.2.7, step 2, from page 11 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating find-equipment-item.

6. Display-text.

This generic task is used to display text to the system operator and wait for acknowledgement. The power plant operating manuals contain some sections that exist exclusively for informational purposes and do not require processing. This information is available to the system operator. This generic task can also be used to display procedure status messages such as "Procedure Step x.x completed." to the system operator.

Figure 3-9 illustrates this generic task by showing operating manual knowledge which must be available to the system operator, but requires no processing by the expert system.

7. Display-diagram.

An operational procedure may require a diagram or flowsheet be displayed. This generic task displays a diagram to the system operator and waits for acknowledgement before

SYSTEM SCOPE

The Liquid Zone Control System is one of several reactivity control devices in the reactor. Control is achieved by varying the amount of demineralized light water contained in the fourteen zone compartments.

The fourteen compartments are contained in six vertically oriented zone control units, penetrating the reactor from above. Each of the two units located on the reactor axial centerline has three compartments, while the outer four assemblies have two compartments each. Water flow out of the bottom of the zones is constant at 0.45 L/s. Level in the zones is changed by varying the inflow from essentially zero to 0.9 L/s.

Figure 3-9 System Scope, from page 4 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating display-text.

continuing.

This generic task is illustrated in Figure 3-10 by showing a procedure step which references a figure.

8. <u>Wait.</u>

This generic task temporarily halts the expert system for a predetermined period of time. It is used primarily when it is necessary to wait for an equipment item to stabilize.

This generic task is illustrated in Figure 3-11 by showing a procedure step requiring that two pressure valves open 5 to 10 seconds after pump has started. The wait generic task is used to temporarily halt the expert system so that the state

Step 1. Disconnect inlet tube fitting on the 14 flow controllers and leave open to the atmosphere. See Figure 1.

Disconnect
Here

FC

To Zones

FI

BUBBLER
HEADER

V 147

Figure 1. Inlet Tube Fittings to Flow Controllers Disconnect Point.

Figure 3-10 Procedure 6.11, step 1, from page 25 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating display-figure.

of these two pressure valves can be verified.

9. <u>Display-current-context.</u>

Context can be defined as the current environment or setting. Within the expert system, context can be thought of as what procedure step is being executed and under what circumstances.

The display-current-context generic task is used to display the current procedure or procedure step being processed. The procedure or procedure step number, name, and description is displayed to the system operator for

Step 9. PV 98 and PV 106 will automatically open 5 to 10 seconds after pump has started. Verify that this occurs. If not, ~~select~~ PV 98 and PV 106 to OPEN, and later correct PV 98 and PV 106 functions.

Figure 3-11 Procedure 5.2.7, step 9, from page 12 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating wait.

confirmation. Execution is initiated upon acknowledgement from the system operator.

Figure 3-12 illustrates this generic task by showing part of

Step 1. Place the system in the shut-down state as described in Section 5.1 herein.

Step 2. Open breakers for pumps P 1, P 2, P 3 and compressors CP 1 and CP 2.

Figure 3-12 Procedure 6.10, steps 1-2, from page 24 of the Liquid Zone Control Operating Manual [Parker and Leclair 1987], illustrating display-current-context.

a procedure for depressurizing zones for flow indicators and flow control repairs. For each procedure step, the display-current-context generic task displays verbatim the procedure step number, name, and description. An operating mode flag in the expert system can be toggled so that only procedure steps that have unsatisfied requirements and need operator action are displayed to the system operator.

10. Link-context.

This generic task allows procedures or procedure steps to be linked together. This link-context generic task along with the other types of link-context generic tasks which follow build the procedure steps and procedures into a semantic network. The type of relation on the arc linking two procedure steps together determine the type of processing the expert system will perform.

Figure 3-13 illustrates this generic task by showing a procedure consisting of 6 steps. Each procedure step would be defined in the expert system using one or several generic tasks. The link-context generic task would be used to link the procedure steps together to ensure that they are done in the proper order. Upon completion of one step, the expert system would begin execution of the next step.

11. Link-context-with-return.

Often, a step in a procedure will require that part or all of another procedure be performed before continuing. This generic task is used to link a procedure step to part or all of another procedure and have the expert system return to its current procedure when done.

5.2.4 Start Circulating Pump

Step 1. Select one pump to OFF.
Step 2. Select a second pump to OFF.
Step 3. Select a third pump to ON.
Step 4. Set flow on FI 74 (0.8 mL/s maximum) by adjusting V, 159.
Step 5. Set flow on FI 75 (0.8 mL/s maximum) by adjusting V 183 .
Step 6. Select PV 2 and PV 31 to AUTO.

Figure 3-13 Procedure 5.2.4, from page 10 of the Liquid Zone Control Operating Manual [Parker and Leclair 1987], illustrating link-context.

Figure 3-14 illustrates this generic task by showing a procedure step requiring that the expert system verify that the system is in shutdown state as described in another procedure.

Step 1. Place the system in the shut-down state as described in Section 5.1 herein.

Figure 3-14 Procedure 6.10, step 1, from page 24 of the Liquid Zone Control Operating Manual [Parker and Leclair 1987], illustrating link-context-with-return.

12. Conditional-link-context.

This generic task is used to conditionally link a procedure step to another procedure or procedure step. The link is in effect only if the expert system finds the conditions that are given are satisfied. The conditions are presented as a list where for each entry in the list it is possible to specify what target equipment and what test to use. Each target equipment may be a single equipment item or several equipment items separated by the addition (+) or subtraction (-) arithmetic operators. The test consists of which slot of the equipment item frame to use, which logical operator to use (<, <= , > , >=, =, <>), and what value to use.

Figure 3-15 illustrates this generic task be showing a procedure step with 3 possible operator actions. Each operator action is represented using one or several generic tasks. The conditional-link-context generic task uses the tank pressure values in the helium storage tank and in the delay tank to determine whether the expert system links to action a, b, or c.

13. <u>Conditional-link-context-with-return.</u>

This generic task is used to conditionally link a procedure step to another procedure or procedure step and have the expert system return to its current procedure when done. The conditions are presented as a list where for each entry in the list it is possible to specify what target equipment and what test to use. Each target equipment may be a single equipment item or several equipment items separated by the addition (+) or subtraction (-) arithmetic operators. The

Step 1. Adjust tank pressures so that the Helium Storage Tank (TK 1) is 300 kPa(g) higher than the Delay Tank (TK 2), as described in (a), (b) and (c) herein.

(a) If TK 1 pressure is above 425 kPa(g) (AI 1064) and TK 2 pressure does not read 300 kPa(g) (minimum) less than TK 1, vent TK 2 by opening V 4, and vent until PI 68 reads 300 kPa(g) lower than TK 1.

(b) If TK 1 pressure is below 425 kPa(g) and TK 2 does not read 300 kPa(g) (minimum) less than TK 1, fill TK 1 by installed method through PRV 24 as follows:

i) back off PRV 24;
ii) open V 220; and
iii) set delivery pressure of PRV 24 (PI 66) to read 300 kPa(g) higher than TK 2 pressure. When this pressure is reached, close V 220.

(c) When helium via PRV 24 is not available, fill TK 1 using bottles (5 required for an empty system) as follows:

i) attach bottled helium to V 150 using a temporary pressure relief valve and set the required delivery pressure;
ii) open V 150;
iii) when TK 1 pressure (AI 1064) reads the set pressure, close the bottle valve, then close V 150.

Figure 3-15 Procedure 5.2.2, step 1, from page 9 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating conditional-link-context.

test consists of which slot of the equipment item frame to use, which logical operator to use (<, <= , > , >=, =, <>), and what value to use.

Figure 3-16 illustrates this generic task by showing a procedure step requiring part of another procedure be completed before proceeding, if the system was shut down for maintenance of flowraters.

> Step 4. If the system was shut down for maintenance of flowrators F 1 to F 14, ensure that Steps 9 to 14 of Section 6.10 are complete before proceeding.
>
> Figure 3-16 Procedure 5.2.1, step 4, from page 8 of the Liquid Zone Control Operating Manual [Parker and Leclair 1987], illustrating conditional-link-context-with-return.

14. Question-link-with-return.

The question-link-with-return generic task is used to link a procedure step to another procedure or procedure step and have the expert system return to its current procedure when done. The question-link-with-return generic task asks the system operator a question and based on the answer the link may or may not take effect. If the system operator's answer does not match the required answer to go to the linked location, then the expert system continues with the next step in the current procedure. If the system operator's answer matches the required answer to go to the linked location, then the expert system will go to the specified procedure and return to its current procedure when done.

Figure 3-17 illustrates this generic task by showing a procedure step asking the system operator to check if water is visible in the flow indicators. If water is present then another procedure must be performed before recommencing this procedure.

Step 5. Check that no water is visible in FI 1 to FI 14 sight glasses. If water is present perform the dry-out procedure described in Section 6.11, then recommence this start-up procedure.

Figure 3-17 Procedure 5.2.1, step 5, from page 9 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating question-link-with-return.

The generic component of the expert system was developed to provide the structure and control for all power plant system operating manuals. Representing the knowledge from another power plant system operating manual in the expert system should not require the generic component to be redeveloped, but rather extended as new tasks are identified. The generic component described above by the generic tasks is implemented as a set of generic frame templates and rule groups which are described in detail in the next three sections.

3.2 Operating Manual Frame Templates

An operating manual frame template has been developed to represent in the expert system each generic task that has been identified (see Section 3.1).

Operating manual frame templates provide a knowledge representation scheme for the power plant system operating manuals. The declarative knowledge contained in the power plant operating manuals is coded in a form understandable to the expert system using the operating manual frame templates. Therefore, the operating manual frame templates can be thought of as a meta language for describing the knowledge found in a power plant system operating manual.

An operating manual frame template is a named collection of slots which when filled describe one generic task. The operating manual frame template which describes a generic task has the appropriate slots necessary to completely define the generic task. Filling the slots of a frame creates what is commonly known as an instance of that frame.

The operating manual frame templates have a slot for procedure number, step, and sequence. The sequence number is used to sequence the operating manual frame templates representing one procedure step. These slots are common to all operating manual frame templates. Filling these slots identify what section of the operating manual the instance pertains to.

Within the expert system, a control frame keeps track of what section of the operating manual is currently being processed by the expert system. Like the operating manual frame templates, the control frame has slots for procedure number, step, and sequence. The expert system will process the instances of the operating manual frame templates with the same procedure number, step, and sequence as the control frame. The control frame slot-values change as various procedures get executed. The control frame points to the section of the operating manual currently being executed. The use of the control frame is similar to the context used in the Mycin [Shortliffe 1986] expert system.

Representing the knowledge from a power plant system operating manual consists of identifying the generic tasks being performed and creating instances of the appropriate operating manual frame templates to represent the generic tasks. The link-context generic tasks are used to link the instances of the operating manual frame templates together. The links and instances of the operating manual frame templates form a semantic network which effectively represents the power plant system operating manual knowledge.

The instances of the operating manual frame templates are structured into a semantic network representing the structure of the system operating manual (see Section 1.3 for a discussion on semantic networks). Figure 3-18 illustrates part of the semantic network that represents procedure 5.2.1, System Preparation prior to Valve Line-up, extracted from the Liquid Zone Control System Operating Manual [Parker and Leclair 1987].

Operational procedures often require that part or all of another procedure be performed. As an example, step 4 of procedure 5.2.1 in Figure 3-18 requires part of another procedure be performed if certain conditions are met. Step 1 of procedure 6.10 requires that all of procedure 5.1 be performed. The semantic network links are used to represent the relations within procedures and the relations amongst procedures.

The semantic network links are implemented using the link-context generic tasks. Each of the link-context generic

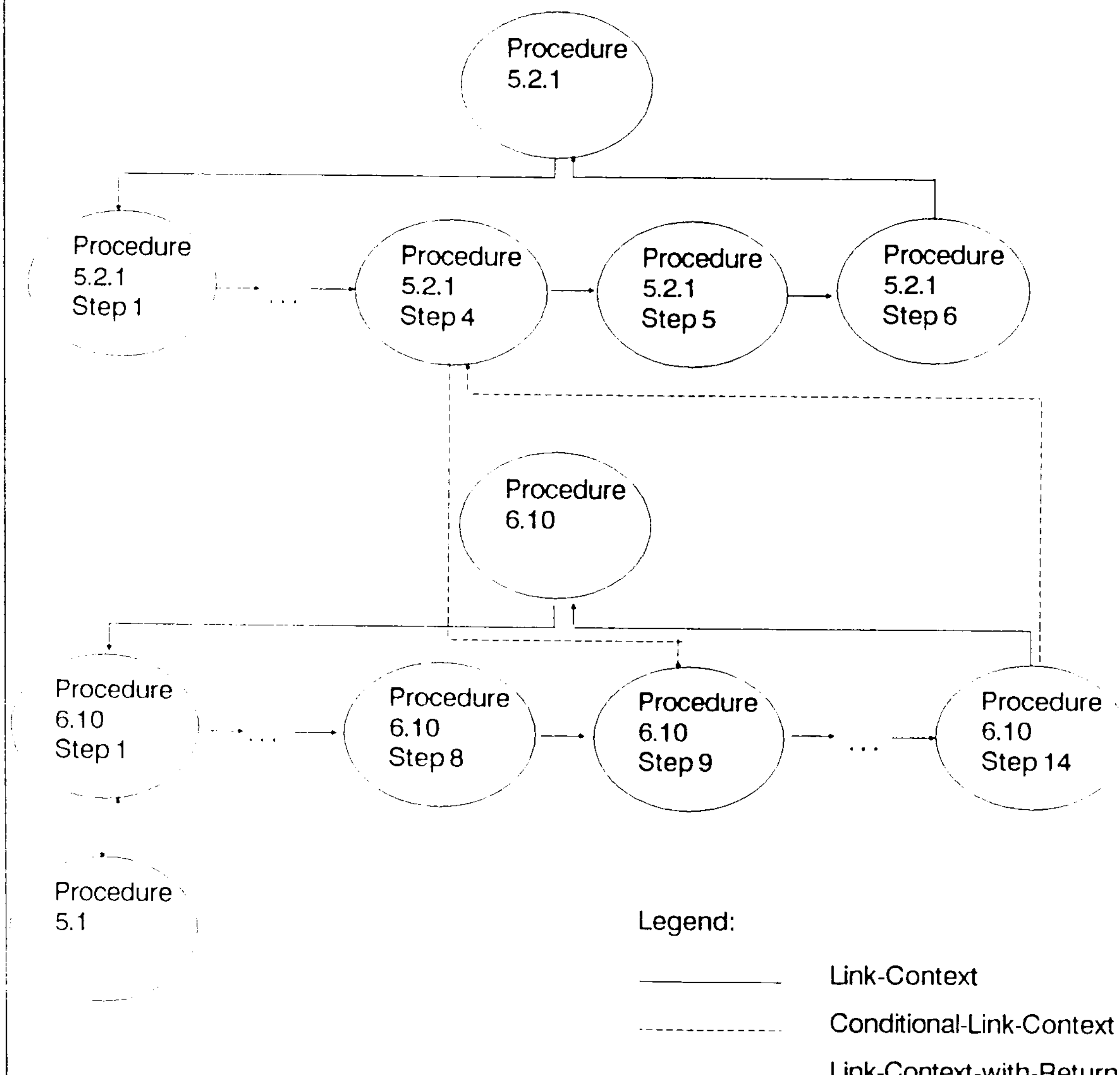


Figure 3-18 Semantic Network Representation of an operational procedure.

tasks described in Section 3.1 implements a different type of link between 2 nodes of the semantic network. The link can be unconditional, based on an answer to a question, or based on equipment states. The type of link used to join 2 nodes together determines the type of processing the expert system will perform.

The link-context generic tasks described in Section 3.1 links a source procedure number, step, and sequence to a destination procedure number, step and sequence. When the source procedure number, step, and sequence matches the control frame, the expert system will link to the destination if the conditions on the link are satisfied. Linking to the destination is done by changing the control frame slot-values to the destination values.

The system operating manual knowledge forms the nodes of the semantic network. For example, a procedure heading, a procedure step, and a window alarm are possible semantic net nodes. A node has a unique label and may be represented using one or more generic tasks.

An example semantic network node can be seen in Figure 3-19 which illustrates the instances of the operating manual frame templates representing semantic network node 5.2.1 step 1 from Figure 3-18. This procedure step requires 3

Instance of:	Display-current-context
Procedure:	5.2.1
Step:	1
Sequence:	1
Text:	Close the valves V 73, V78, and V 103 to isolate the balance header.

Instance of:	Requirement-and-logic
Procedure:	5.2.1
Step:	1
Sequence:	2
Equipment:	V 73, V 78, V 103
Property:	state, state, state
Operator:	=, =, =
Value:	Closed, Closed, CLosed

Instance of:	Display-text
Procedure:	5.2.1
Step:	1
Sequence:	3
Text:	5.2.1 Step 1 Completed.

Instance of:	Link-context
Procedure:	5.2.1
Step:	1
Sequence:	4
To Procedure:	5.2.1
To Step:	2
To Sequence:	1

Figure 3-19 Procedure 5.2.1, step 1, from page 8 of the Liquid Zone Control System Operating Manual [Parker and Leclair 1987], illustrating operating manual frame templates.

valves be in a specific state. The expert system will process the instances of the operating manual frame templates representing the procedure step in order by sequence number.

The generic tasks used to described a semantic network node determine the processing the expert system will perform at this node. The procedure step in Figure 3-19 was broken down into 4 generic tasks, each generic task being represented by an instance of the appropriate operating manual frame template: display-current-context, equipment-state-requirement-with-and-logic, display-text, and link-context. The display-current-context will display the procedure step information to the system operator. The equipment-state-requirement-with-and-logic generic task will verify that the 3 valves are in the closed state. The display-text is used to display a procedure step completed message to the system operator. The link-context links this procedure step to the next procedure step.

Operating manual frame templates have been developed for all the generic tasks identified in Section 3.1.

3.3 Equipment Frame Templates

Equipment frame templates are used to represent the physical system described in the system operating manual. This consists of all the equipment items and their relationships to one another. Generally, an equipment frame template will exist for each type of equipment item in the system. Each equipment frame template has the appropriate slots for the type of equipment item being considered.

Instances of the appropriate equipment frame template are made for each equipment item in the system by filling the frame slots. Figure 3-20 illustrates an instance of the valve equipment frame template describing valve 160 (V160) and an instance of the pump equipment frame template describing pump 1 (P1). Certain slot-values such as the state of an equipment item could be updated by the control computer in real time.

Relations allow the building of a semantic network of equipment frames (see Section 1.3 for a discussion on relations and semantic networks). Several relations with their respective inverse relation have been used for describing the physical system in greater detail. These relations transform the equipment frames from an equipment

Instance of:	Valve
Equip id:	V160
Description:	Drain on H2O Return Header
Belongs to:	Associated Piping
State:	Open

Instance of:	Pump
Equip id:	P1
Description:	System Pump 1
Belongs to:	Demineralized water system
State:	Off
Breakers:	Closed

Figure 3-20 Equipment frame templates.

list to a more useful and accurate hierarchical representation of the complete physical system. Certain relations provide for inheritance of information (see Section 1.3 for a discussion on inheritance).

Note that defining an instance of a relation implies the existence of an inverse relation. A common example is the father-of and son-of relation. Stating that John is the father-of Dave implies Dave is the son-of John. Thus, relations always come in pairs. Some expert systems shells will automatically create the inverse relation when a relation is defined.

The consists-of relation allows the identification of all the equipment items which make up a particular system or sub-system. For example, in the Liquid Zone Control System [Parker and Leclair 1987], the helium cover gas subsystem consists-of helium tank 1 (TK1). The inverse relation is named belongs-to and it gives what system or sub-system an equipment item belongs to. This is not an inheritance relation. Figure 3-21 illustrates the semantic network defined by the consists-of relation for the Liquid Zone Control System [Parker and Leclair 1987].

The is-a relation and its inverse relation kinds are commonly used. This is an inheritance relation and it is used to link general concepts together. For example, we may want to link the equipment item type valve to the general frame named equipment. We would do this by saying valve is-a equipment. Defining an instance of the is-a relation implies the existence of the inverse relation kinds. Therefore valve is-a equipment implies equipment kinds valve. Linking all the equipment types to a parent frame named equipment allows the slots common to all equipment types to be described once in the frame equipment, and then inherited by all equipment types. Figure 3-22 illustrates the semantic network defined by the is-a relation for the Liquid Zone Control System [Parker and Leclair 1987].

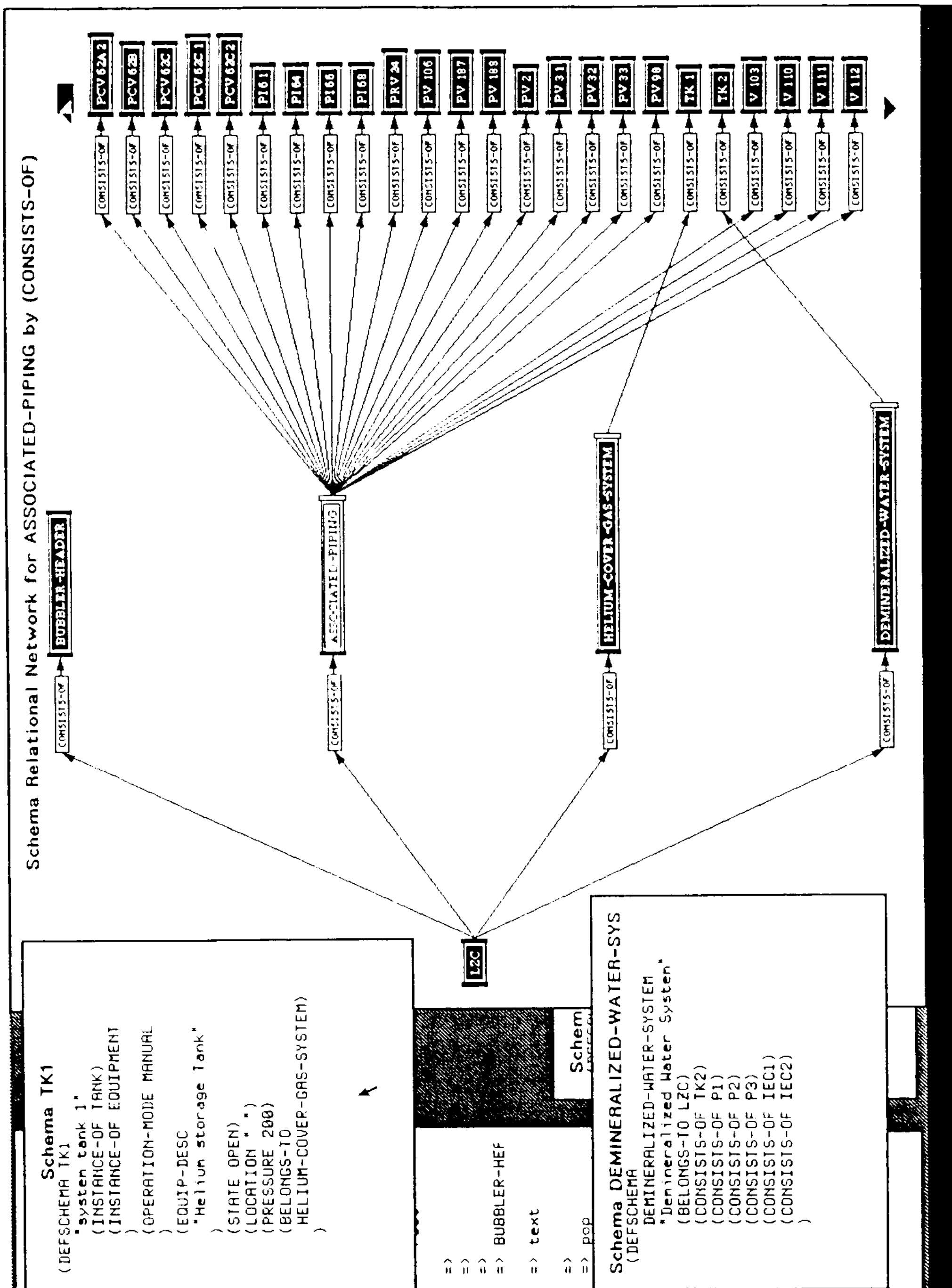


Figure 3-21 Equipment semantic network defined by the consists-of relation.

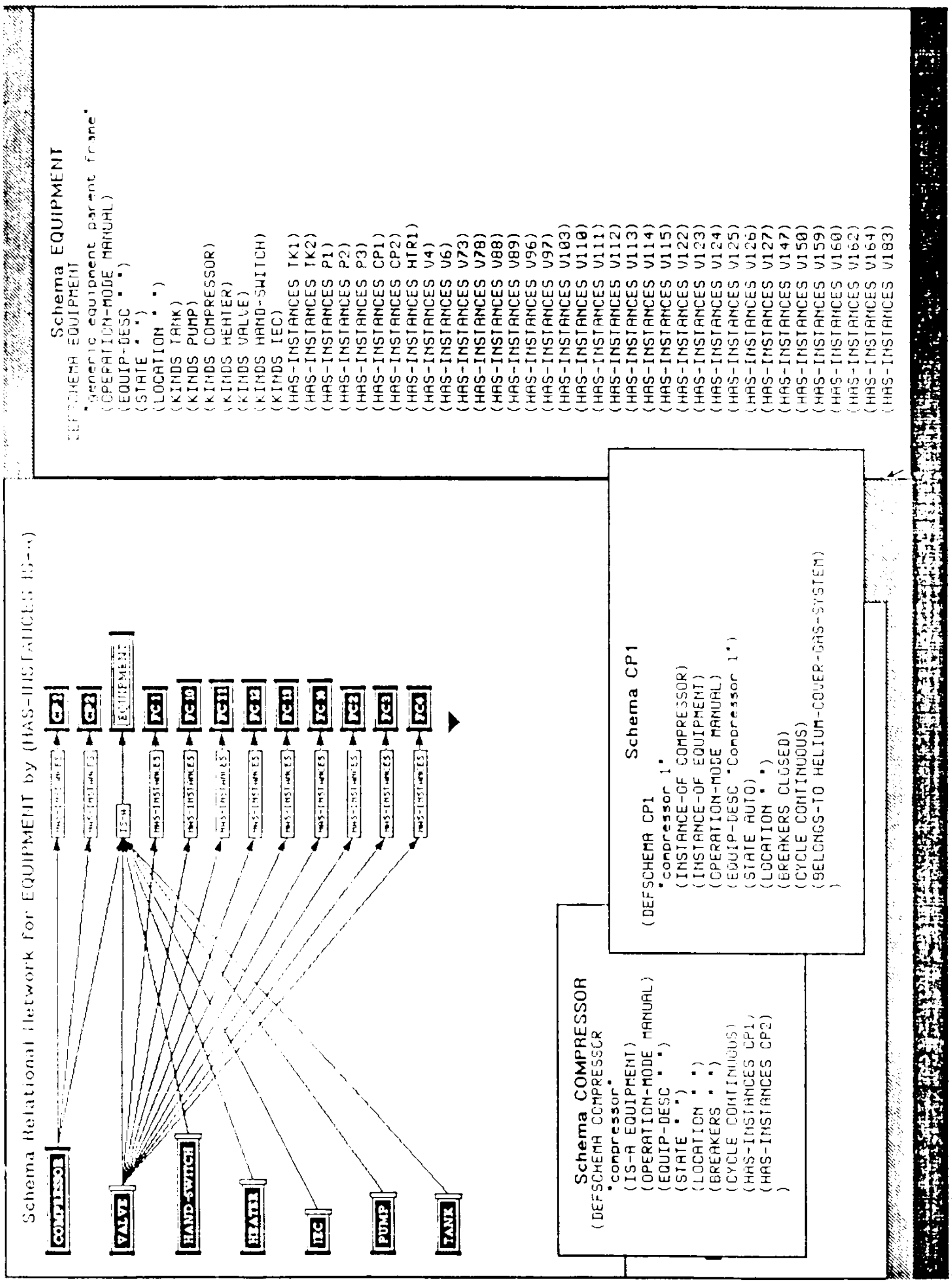


Figure 3-22 Equipment semantic network defined by the is-a and has-instances relations.

The has-instances relation shows that a frame is one of something. For example, an equipment item type such as pressure valve has many specific instances, such as pressure valve 68 (PV68). The inverse relation instance-of gives for a specific equipment item the type of equipment item. The has-instances is an inheritance relation. Figure 3-22 illustrates the semantic network defined by the has-instances relation for the Liquid Zone Control System [Parker and Leclair 1987].

Equipment frame templates and their relations have been developed for the Liquid Zone Control System [Parker and Leclair 1987].

3.4 Rule Groups

For each generic task that has been identified (see Section 3.1), a rule group has been developed to oversee the execution of this task. A rule group is simply a group of generic rules responsible for performing the action described by one of the generic tasks.

The rule groups form one level of control of the expert system. Both these generic rules and the inference engine process the declarative knowledge contained in the power plant system operating manuals.

These generic rules have no operating manual system specific knowledge within them. The knowledge the generic rules contain consists of how a power plant operating manual is structured and how to interpret and exploit the knowledge found within them in order to make logical decisions. These generic rules represent the knowledge possessed by the power plant system operator about how to use the power plant system operating manuals, and not the power plant system operating manual knowledge itself.

Just like the human power plant system operator, the rule groups must interpret the operational procedures contained in the power plant system operating manuals and, if necessary, recommend an appropriate course of action. The rule groups ensure that a procedure is correctly performed by ensuring that all the generic tasks defining the procedure are correctly performed. The rule groups decide which sections of the procedure are applicable under the current circumstances, and then leads the system operator through these sections, ensuring that they are properly performed. In this way, the rule groups act as an assistant to the power plant system operator.

It is interesting to note that because of the data driven approach and the way forward chaining expert systems fire rules opportunistically, the expert system may appear to perform more than one task at a time. For example, the expert system may be leading the system operator through an operational procedure and at the same time be monitoring the status of several equipment items.

The rule groups process the declarative knowledge contained in the system operating manuals. The operational procedures and other knowledge contained in the operating manuals are broken down into generic tasks. The generic tasks are represented in the expert system using operating manual frame templates; the rule groups process these operating manual frame templates (see Figure 3-3).

The rules groups must process the declarative knowledge in the same sequence as given in the operating manual. Rule groups will process operating manual frame templates with the same procedure number, step, and sequence as the control frame (see Section 3-2 for a discussion on the control frame). The control frame slot-values are changed by the rule groups as various procedures get executed.

An example using the rule group responsible for the monitor-equipment-item generic task (see Section 3.1, generic task 4) is given below. Figure 3-23 illustrates a procedure step requiring that valve 160 be left open during the repair or

Step 8. Leave V 160 open during the repair or dry-out. If isolation is required for maintenance, it can now be applied. If dry-out is the next requirement, proceed to Section 6.11, Step 1.

Figure 3-23 Procedure 6.10, step 8, from page 24 of the Liquid Zone Control Operating Manual [Parker and Leclair 1987], illustrating request for monitoring.

dry-out. This procedure step requires the monitoring of valve 160.

The request for monitoring of valve 160 is represented using an instance of the monitor-equipment-item operating manual frame template. The equipment item valve 160 is represented in the expert system using an instance of the valve equipment frame template. The control frame slots contain the procedure number, step, and sequence currently being processed by the expert system. Figure 3-24 illustrates the instance of the monitor-equipment-item frame template describing this monitoring request, the instance of the valve frame template representing valve 160, and the control frame.

Figure 3-25 depicts the rules in pseudo code for the monitor-equipment-item rule group. This rule group consists of 4 rules: start-monitoring, monitor, display-monitor-message, and end-monitoring. A description of each rule is a)

Instance of:	Monitor-equipment-item
Procedure:	6.10
Step:	8
Sequence:	1
Equipment:	V 160
Property:	state,
Operator:	=
Value:	Open
Stop monitoring:	6.10 step 14

b)

Instance of:	Valve
Equip id:	V160
Description:	Drain on H20 Return Header
Belongs to:	Associated piping
State:	Open

c)

Instance of:	Control frame
Description:	Controls processing
Procedure:	6.10
Step:	8
Sequence:	1

Figure 3-24 Frame templates processed by rule group. a) Instance of the monitor-equipment-item frame template. b) Instance of the valve equipment frame template. c) Control frame.

given below.

The monitor-equipment-item rule group becomes active when a request for monitoring is invoked by the expert system as required by an operational procedure. The start-monitoring rule is fired when the procedure number, step, and sequence on the instance of the monitor-equipment-item frame template

Rule: Start-monitoring
If (Instance of Monitor-equipment-item frame) and
(Control.procedure = Monitor-equipment-item.procedure) and
(Control.step = Monitor-equipment-item.step) and
(Control.sequence = Monitor-equipment-item.sequence)
Then
Assert fact (Monitor-equipment-item.equipment, Monitor-equipment-item.property, Monitor-equipment-item.operator,Monitor-equipment-item.value, Monitor-equipment-item.stop-monitoring).

Rule: Monitor (Priority High)
If (fact (Monitor-equipment-item.equipment, Monitor-equipment-item.property, Monitor-equipment-item.operator,Monitor-equipment-item.value, Monitor-equipment-item.stop-monitoring)) and
(property of instance of equipment not operator value)
Then
Assert fact(Equipment, Failed monitoring test).

Rule: Display-monitor-message
If (fact(Equipment, Failed monitoring test))
Then
Display warning message to system operator
Wait for acknowledgement.

Rule: End-monitoring
If (fact (Monitor-equipment-item.equipment, Monitor-equipment-item.property, Monitor-equipment-item.operator,Monitor-equipment-item.value, Monitor-equipment-item.stop-monitoring)) and
(Control.procedure = Monitor-equipment-item.stop-monitoring-procedure) and
(Control.step = Monitor-equipment-item.stop-monitoring-step)
Then
Retract fact (Monitor-equipment-item.equipment, Monitor-equipment-item.property, Monitor-equipment-item.operator,Monitor-equipment-item.value, Monitor-equipment-item.stop-monitoring).

Figure 3-25 Monitor rule group

matches the values on the control frame. This rule puts a monitor fact in the knowledge-base indicating the equipment item, the monitoring test, and the end procedure step. The monitor test consists of which property of the equipment item to use, which logical operator to use (<, <= , > , >=, =, <>), and what value to use. These parameters are obtained from the instance of the monitor-equipment-item frame template representing this particular procedure.

The monitor rule has been defined as having high priority. The expert system continuously performs the monitoring test on the instance of the equipment frame template representing the equipment item. If the equipment item passes the test, the expert system goes on to the next step in the operational procedure. If the equipment item fails the test, the display-monitor-message rule is invoked.

The display-monitor-message rule displays a warning message to the system operator indicating that the specified equipment item has failed the monitoring test. The system operator must acknowledge this warning message before continuing with the procedure. Corrective action may be initiated by the system operator.

The end-monitoring rule is fired when the end procedure step contained on the monitor fact is encountered. Removing the fact effectively stops the monitoring of the equipment item.

Rule groups have been developed for all the generic tasks identified in Section 3.1.

Chapter 4 EXAMPLES FROM OTHER OPERATING MANUALS

The Liquid Zone Control System [Parker and Leclair 1987] used in the examples and in the prototype expert system is one of approximately 200 systems at the Point Lepreau Nuclear Generating Station. Of the 200 system operating manuals, approximately 10 are considered more important than the rest. The approach to representing power plant system operating manual knowledge described in Chapter 3 is applicable to all the system operating manuals at Point Lepreau.

Figure 4-1 illustrates part of the operating manual index used at Point Lepreau. An index similar to this would be used as the first screen of the expert system. Using the concepts developed in this thesis, an expert system could be developed which would contain the knowledge from all these operating manuals.

The following section provides additional examples of knowledge representation from a second system operating manual: Feedwater Heater and Separator Drains [Dean 1986].

as of Aug. 29/88

OPERATING MANUAL INDEX

OPERATING MANUAL #	REV. #	DATE ISSUED	DESCRIPTION
33100/ 33370	Rev. 4/9	Oct. 3/86	Main Heat Transport System/Autoclave System
33320	Rev.3/12	88-04-14	HT Pressure & Inventory Control System
33330	Rev. 3/3	June 1/87	PHT Coolant Storage Transfer & Recovery Circuit
33340	Rev. 3/1	Aug. 24/88	H.P. Pump Gland Seal Cooling System
33350	Rev. 2/4	Aug. 25/88	Heat Transport Purification
33410	Rev. 3	July 18/88	Shutdown Cooling System
33530	Rev. 0	June 10/82	R/B N_2 Addition System
33540	Rev. 1/3	Oct. 3/86	H.T. Hydrogen Addition System
33810	Rev. 2/4	87-10-01	D_2O Collection Sys.
34110	Rev. 0/2	May 9/85	Shield Cooling Sys.
34310/ 63431	Rev. 2/8	Apr. 25/88	Dousing System
34320/ 63432	Rev.1/11	July 28/88	Emergency Core Cooling
34410	Rev. 2/1	Oct. 20/87	Spent Fuel Bay Cooling & Purification Sys.
34510	Rev. 3	July 24/87	Resin Transfer Sys.
34610/ 63461	Rev. 0/3	Apr. 25/88	Emergency Water Supply System (EWS/BMW)
34700	Rev. 3/1	July 10/86	Liquid Injection Shutdown Sys.

Figure 4-1 Subset of the Operating Manual Index (approximately 10% of the total for Point Lepreau).

4.1 Feedwater Heater and Separator Drains

An appendix common to many of the power plant system operating manuals is the Valve and Handswitch List. This appendix provides a list of system valves and handswitches with columns relating to the system states where the valve state (open, closed, throttled, variable, etc.) and handswitch state (on, off, standby, auto, etc.) can be listed [Johnson 1988].

Figure 4-2 illustrates part of the Valve and Handswitch List from the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986]. This particular list is 25 pages long and contains approximately 500 valve and handswitch states for both the shutdown and normal operating state.

The Valve and Handswitch List contains the generic task equipment-state-requirement-with-and-logic (see Section 3.1, generic task 1) which is represented in the expert system using the equipment-state-requirement-with-and-logic operating manual frame template.

NUCLEAR OPERATIONS

ENERGIE NB POWER POINT LEPREAU G.S | OPERATING MANUAL | FEEDWATER HEATER AND SEPARATOR DRAINS | 1 | 43130

APPENDIX I VALVE AND HANDSWITCH LIST

For the following, assume "NORMAL OPERATING STATE" to occur at 100% MCR with both H.P. heater drains pumps in service and their handswitches selected to "ON", and HS115 and HS116 selected to "LCV115A" and "LCV116A" respectively.

EQUIP.NO.	EQUIPMENT DESCRIPTION	SHUTDOWN STATE POSITION	NORMAL OPERATING STATE POSITION
LCV1	LP3B F/W HTR Bypass level control valve	CLOSED	CLOSED
LCV2	LP3A F/W HTR " " " "	CLOSED	CLOSED
LCV3	LP3B F/W HTR Cascade level control valve	CLOSED	VARIABLE
LCV4	LP3A F/W HTR " " " "	CLOSED	VARIABLE
LCV13	LP2B F/W HTR Bypass level control valve	CLOSED	CLOSED
LCV14	LP2A F/W HTR " " " "	CLOSED	CLOSED
LCV15	LP2B F/W HTR Cascade level control valve	CLOSED	VARIABLE
LCV16	LP2A F/W HTR " " " "	CLOSED	VARIABLE

Figure 4-2 Appendix 1, from page 76 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986], illustrating Valve and Handswitch List.

Figure 4-3 illustrates a procedure step which references the Valve and Handswitch appendix. Figure 4-4 illustrates the semantic network for this procedure step and the instances of the frame templates which represent this network. The expert system will process this list of equipment items by verifying that each equipment item is in the correct state for the current system state. Any equipment items not in the correct state will be brought to the system operator's attention.

Shutdown State (Cont'd)

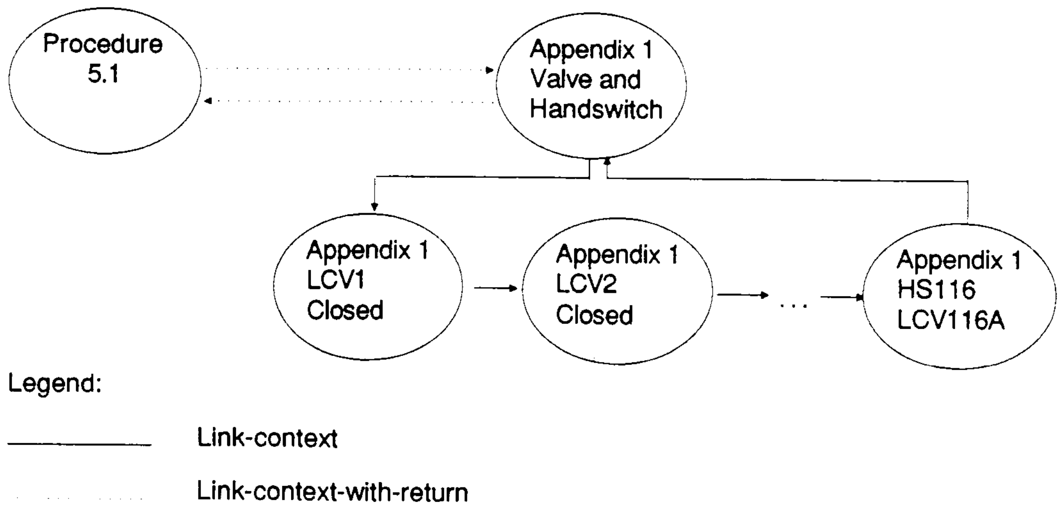
The status of all valves and handswitches in the shutdown state is shown in Appendix I. (Valve and Handswitch List).

Figure 4-3 Procedure 5.1, from page 10 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986], illustrating referencing an appendix.

All the power plant system operating manuals contain detailed approved operational procedures [Johnson 1988]. Figure 4-5 illustrates the required operator action following Alarm CI 1348 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986]. The operator action in this case consists of performing action 1, 2, or 3 based on certain conditions.

The operator action described in Figure 4-5 contains the generic task conditional-link-context (see Section 3.1, generic task 12) which is represented in the expert system using the conditional-link-context operating manual frame template. Figure 4-6 illustrates these operator actions as a semantic network and shows the instances of the frame templates which represent this network.

Not only can operational procedures reference part or all of another procedure within the same operating manual, but



Instance of:	Display-current-context
Procedure:	5.1
Step:	1
Sequence:	1
Text:	The status of all valves and handswitches in the shutdowm state is shown in Appendix 1 (Valve and Handswitch List).

Instance of:	Link-with-return
Procedure:	5.1
Step:	1
Sequence:	2
To Procedure:	Appendix 1
To Step:	Shutdown
To Sequence:	1

Instance of:	Display-current-context
Procedure:	Appendix 1
Step:	Shutdown
Sequence:	1
Text:	Appendix 1 Valve and Handswitch List

Instance of:	Requirement-and-logic
Procedure:	Appendix 1
Step:	Shutdown
Sequence:	2
Equipment:	LCV1
Property:	state
Operator:	=
Value:	Closed

...

Instance of:	Requirement-and-logic
Procedure:	Appendix 1
Step:	Shutdown
Sequence:	521
Equipment:	HS116
Property:	state
Operator:	=
Value:	LCV116A

Figure 4-4 Semantic network and instances of frame templates for procedure 5.1 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986].

Operator Action:

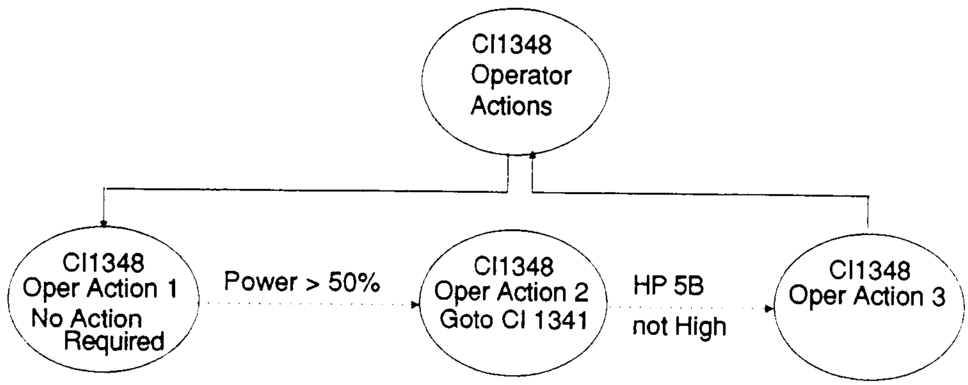
1. If power level is below 50% MCR, no action is required.
2. If HP SB level is high, see action for CI 1341.
3. If neither of above, check local sight glass LG 119 and operation of cascade and bypass control valves - LCV 115 (A or C) and LCV 113, respectively.
   (a) Select alternate cascade level control valve LCV115A or C if required.

Figure 4-5 Operator Action following Alarm CI 1348 from page 62 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986].

occasionally they reference procedures from other operating manuals. When an operational procedure makes reference to a section of another operating manual, the expert system would have that knowledge available.

Figure 4-7 illustrates a procedure step from the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986] which references another system operating manual. This procedure step describes an automatic action to the system operator.

The question-link-with-return generic task (see Section 3.1, generic task 14) could be used in this case to ask the



Legend:

——————— Link-context

Conditional-link-context

Instance of:	Conditional -link-context
Procedure:	CI1348
Step:	1
Sequence:	1
Equipment:	HTR05B
Property:	power
Operator:	<
Value:	50 %
To Procedure:	CI1348
To Step:	Menu
To Sequence:	1

Instance of:	Link-context
Procedure:	CI1348
Step:	1
Sequence:	2
To Procedure:	CI1348
To Step:	2
To Sequence:	1

Instance of:	Conditional-link-context
Procedure:	CI1348
Step:	2
Sequence:	1
Equipment:	HP5B
Property:	level
Operator:	=
Value:	high
To Procedure:	CI1341
To Step:	1
To Sequence:	1

Instance of:	Link-context
Procedure:	CI1348
Step:	2
Sequence:	2
To Procedure:	CI1348
To Step:	3
To Sequence:	1

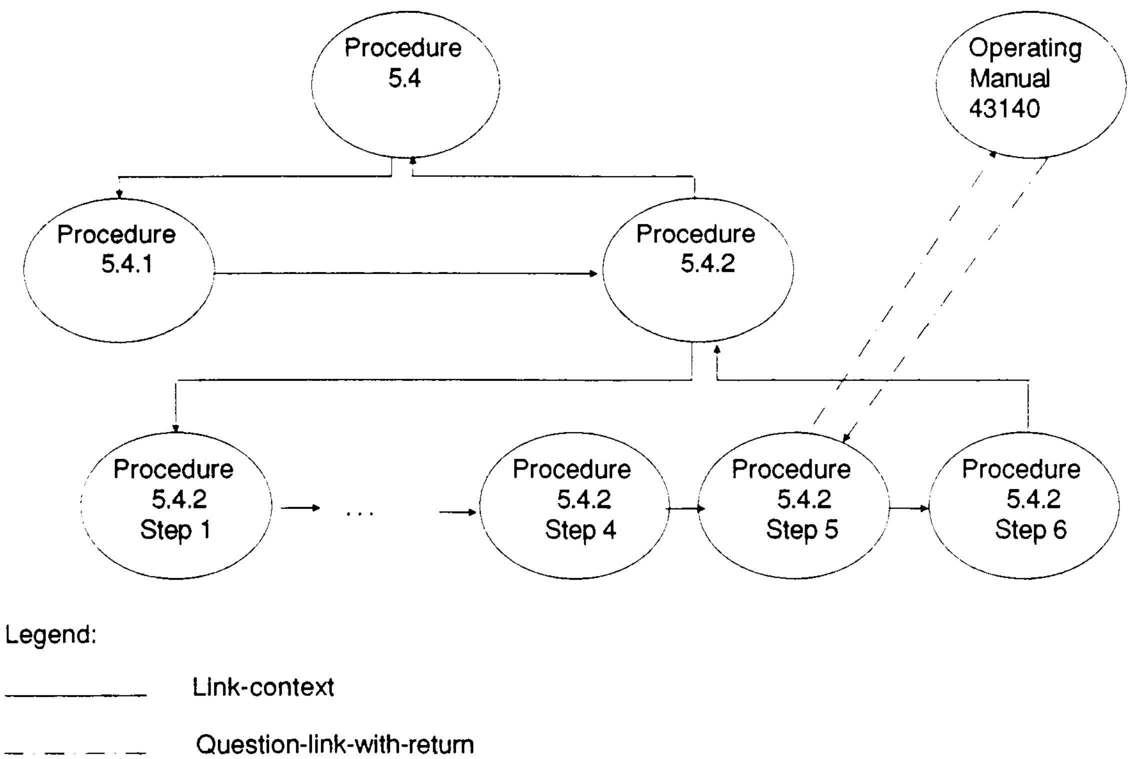
Figure 4-6 Representation of operator action required for Alarm CI 1348 of the Feedwater Heater and Separator Drains System Operator Manual [Dean 1986].

5) The H.P. heater shell start up vents will automatically open at 10% MCR decreasing (Ref. O.M. 43140).

Figure 4-7 Procedure 5.4.2, step 5, from page 20 of **the Feedwater Heater and Separator Drains System** Operating Manual [Dean 1986], illustrating referencing another operating manual.

system operator whether to link to the specified operating manual for more information. Upon returning, the expert system will continue with the next procedure step. Figure 4-8 illustrates the semantic network for this procedure step and the instances of the frame templates which represent this network.

The approach described above can be applied to all power plant system operating manuals. The same approach can be applied to the more general problem of representing and applying the knowledge from any well structured operating manual with significant complexity, such as that found in process systems and power plants. For example, a process system in a pulp mill would be a potential candidate for representation within the expert system.



Instance of:	Display-current-context
Procedure:	5.4.2
Step:	5
Sequence:	1
Text:	The H.P. heater shell start up vents will automatically open at 10% MCR decreasing (Ref. O.M. 43140).

Instance of:	Question-link-with-return
Procedure:	5.4.2
Step:	5
Sequence:	2
To Procedure:	OM 43140
To Step:	Menu
To Sequence:	1
Question:	Do you want more information?
Accept:	Yes

Instance of:	Link-context
Procedure:	5.4.2
Step:	5
Sequence:	3
To Procedure:	5.4.2
To Step:	6
To Sequence:	1

Figure 4-8 Semantic network and instances of frame templates for required operator action following alarm CI 1348 of the Feedwater Heater and Separator Drains System Operating Manual [Dean 1986].

Chapter 5 PROTOTYPE IMPLEMENTATION

An initial meeting with Point Lepreau staff identified the functional requirements of the prototype Interactive Operating Advisor. The Interactive Operating Advisor is not a monitoring system, but rather an intelligent assistant who is capable of aiding the system operator in identifying and applying the sections of the system operating manual which are currently applicable. It takes the passive data contained in the system operating manuals and utilizes it the way a human expert would; i.e. interprets it and decides how to exploit it in order to solve the problem at hand. The primary objective is to make the operating manual knowledge more readily accessible and effective for the power plant system operators.

A requirement made by Point Lepreau staff is that ultimate control and decisions must remain the system operator's responsibility. The Interactive Operating Advisor acts only as an assistant to the system operator. At any time the system operator can stop execution of the current procedure and return to the controlling menu at this level.

The Interactive Operating Advisor provides a menu interface similar to the standard index contained in all power plant system operating manuals (see Section 1.2, Figure 1-1). The expert system presents a menu to the system operator and waits for selection. It is the system operator who initiates the execution of a procedure by making a selection from the menu. Once an operational procedure has been initiated, the expert system will ensure that it is properly executed.

An appropriate knowledge representation structure was determined by studying the Liquid Zone Control system operating manual of the Point Lepreau nuclear generating station and the architecture of successful expert systems. Two prototype expert systems were built according to the design described in Chapter 3 using different expert system shells. Each shell provided one or several knowledge representation schemes and the pros and cons of each are presented.

The interface to the power plant control computer, which can ultimately provide plant status and parameters to the expert system, was simulated by manually setting the initial states of all objects referred to by the Liquid Zone Control System.

The first prototype was built using the FLOPS [Siler and Tucker 1986] expert system shell. After approximately 6 weeks of development, it became evident that this shell did not have the advanced capabilities needed to implement this expert system.

The second expert system prototype was built using the ART [Clayton 1987] expert system shell. This product met all the requirements and 4 months of development resulted in a prototype Interactive Operating Advisor.

The following sections describe the development for each of the 2 expert system prototypes.

5.1 FLOPS

Fuzzy LOgic Production System (FLOPS) is an inference engine supposedly based on the forward chaining OPS5 system developed at Carnegie-Mellon University. The FLOPS expert system shell provides several features which distinguish it from other expert system shells. First, FLOPS supports fuzzy numbers, fuzzy sets, and fuzzy logic. Second, FLOPS has the ability to emulate parallel computation by firing several rules effectively in parallel. Fuzzy arithmetic and parallel computation were, however, not considered necessary for developing the Interactive Operating Advisor.

Data is represented in FLOPS using memory elements. Memory elements are similar to frames. The declaration of a memory element sets up a template which will be used in storing actual data, called instances of the element [Siler and Tucker 1986].

Implementing the operating manual frame templates and the equipment item frame templates using memory elements quickly came to a dead end. The knowledge representation capability of memory elements is limited and restricted to objects and instances of objects. Inheritance, multi-valued slots, daemons, relations amongst the objects, and other advanced features are not supported.

Rules in FLOPS consist of a rule priority, a left hand side which specifies the conditions under which the rule may be fired, and a right hand side which specifies the actions to be taken when the rule is fired. FLOPS rule syntax provided the flexibility needed to implement the rule groups as described in Section 3.4.

When more than one rule is fireable, FLOPS picks the rule with the highest confidence factor and places the remaining rules on the rule stack. If FLOPS reaches a point where no more rules are fireable, it backtracks by popping the rule stack. In this way, FLOPS is like the artificial programming language Prolog [Clocksin and Mellish 1984], and differs substantially from OPS5.

When backtracking, FLOPS handles memory in its own unique way which caused some problems and surprises. Modifying or deleting a memory element does not effect rules that have been put on the rule stack for backtracking. FLOPS remembers what made the rule fireable and will bring the instances of memory elements back to life for the backtracked rule. This made it difficult to implement the rule groups so that they fire properly. If a rule is put on the rule stack, it will eventually fire no matter what is done to the instances of memory elements which made the rule fireable.

Another problem caused by FLOPS backtracking memory management is that it often leads to multiple instances of the same memory element being created. This occurs due to the fact that, when FLOPS puts a rule on the rule stack, it also puts a snapshot of the memory element used to match the rule's left hand side. If this rule is not immediately fired and another rule changes this memory element, a new memory element will be created. Two active versions of the same memory element will exist if the stacked rule modifies the memory element. The programmer is responsible for writing rules to delete the unwanted duplicate memory elements.

The FLOPS prototype expert system was developed using a 10 MHZ 286 AT personal computer with a 40 MB hard drive with 38 ms access time. The prototype expert system ran slow on this machine. The prototype was then ported to a 16 MHZ 386 Compaq Deskpro. Although response time improved on this machine the prototype still seemed to spurt out one line at a time. Painting the entire screen was extremely slow. FLOPS rules are compiled to pseudocode and an interpreter executes the pseudocode. This is not acceptable with the large knowledge-base required for the power plant system operating manuals.

FLOPS seems to have been developed for relatively small expert systems. The FLOPS shell requires 132 KB and limits working memory to 128 KB. Deleting a memory element does not clear this memory element from memory. FLOPS merely flags deleted memory elements, and does not attempt to recover any space. The modification of a memory element actually marks the memory element as deleted and creates a new memory element with the modified valves. This is necessary to support the backtracking as described above.

Because of the previous points mentioned, the expert system is very limited in size. Even a small expert system which is run a few times will eventually halt because of a storage overflow. The prototype expert system which consisted of 35 rules and 30 KB of data will run for approximately 5 minutes before a storage overflow is reported.

After approximately 6 weeks of development, it was concluded that FLOPS was not a good choice for implementing the Interactive Operating Advisor.

5.2 ART

The second prototype expert system was built using the Automated Reasoning Tool (ART) [Clayton 1987] expert system shell on a Symbolics Lisp machine.

The expert system developer's interface to ART is through the ART Studio which consists of three parts: the browser, the execution monitor, and the artist graphics studio. The browser is the interface for examining and altering the knowledge-base which consists of facts, schemata, and rules. The execution monitor is used to debug and run the expert system. The artist graphics studio is a tool for building menus and graphics. The ART studio proved to be an effective and productive environment during all phases of system development.

ART is flexible and powerful in knowledge representation. The declarative knowledge contained in the power plant system operating manuals can be represented in ART using facts and schemata. Facts and schemata can be used to represent just about any type of knowledge.

In ART, a fact is a proposition known to be true for a certain time frame. For example a fact may be the proposition "the car is red" and the time frame "at this particular time". Facts are fundamental pieces of knowledge which are used to pass information amongst the rules.

A schema is a named collection of facts which describe an object. Schemata (plural of schema) are very similar to frames. ART's schema system provides many advanced features such as multi-valued slots, daemons, inheritance, and natural language support. ART also provides 2 standard inheritance relations: is-a with it's inverse kinds, and instance-of with it's inverse has-instances. The expert system developer can also easily define relations with or without inheritance. These relations can be used to link the schemata into a hierarchy or network.

The facts, relations, and schemata organize the knowledge-base into a semantic net (see Section 1.3 for a discussion on semantic nets). A semantic net is a means of representing knowledge by defining objects in terms of their relations to one another [Clayton 1987]. In ART, the schemata form the nodes of the semantic net and the relations form the links of the semantic net. As discussed in Chapter 3, the operating manual knowledge can be effectively represented using semantic nets.

ART's inference engine supports both forward and backward chaining. The operating manual procedures are driven by the power plant system parameters. This means that the forward chaining, data driven approach is best suited for the Interactive Operating Advisor expert system. Therefore the prototype expert system uses the forward chaining approach.

In ART, a rule consists of a left hand side, a separator, and a right hand side. The left hand side of a rule consists of one or several conditions which must be met before the rule can be considered for firing. The conditions are specified using patterns and tests. A pattern describes the facts that are necessary for a match to occur. ART's pattern matching language allows for literals, variables, wildcards, logical operators, and much more. Tests are used to execute a Lisp function as part of the conditions of a rule. A test would be used when the condition cannot be expressed in ART's pattern matching language. Using both patterns and tests provides the expert system developer with the capability of specifying virtually any type of conditions for left hand side pattern matching.

The right hand side of a rule describes the actions that are to be performed when the rule is fired. Possible actions include asserting and retracting facts, asserting and retracting schemata, modifying facts, modifying slots of schemata, and user interface actions.

ART's forward chaining inference engine is purely data driven. An agenda keeps track of all the rules which have their left hand side conditions satisfied. ART evaluates the agenda and fires the rule with the highest priority. After firing the rule the agenda is revised and the cycle starts over again. If the conditions of a rule on the agenda are no longer satisfied, then the rule is no longer fireable and is removed from the agenda. In this way, ART is very similar to OPS5 [Forgy 1979] and differs substantially from FLOPS. In ART, there is no concept of backtracking while forward chaining as in FLOPS.

ART rules can be assigned a salience which determines the priority or importance of the rule when competing with other rules. The rule with the highest salience is selected from the agenda for firing. Rule salience can be used to stratify the rule base into groups of rules which have different priorities [Clayton 1987].

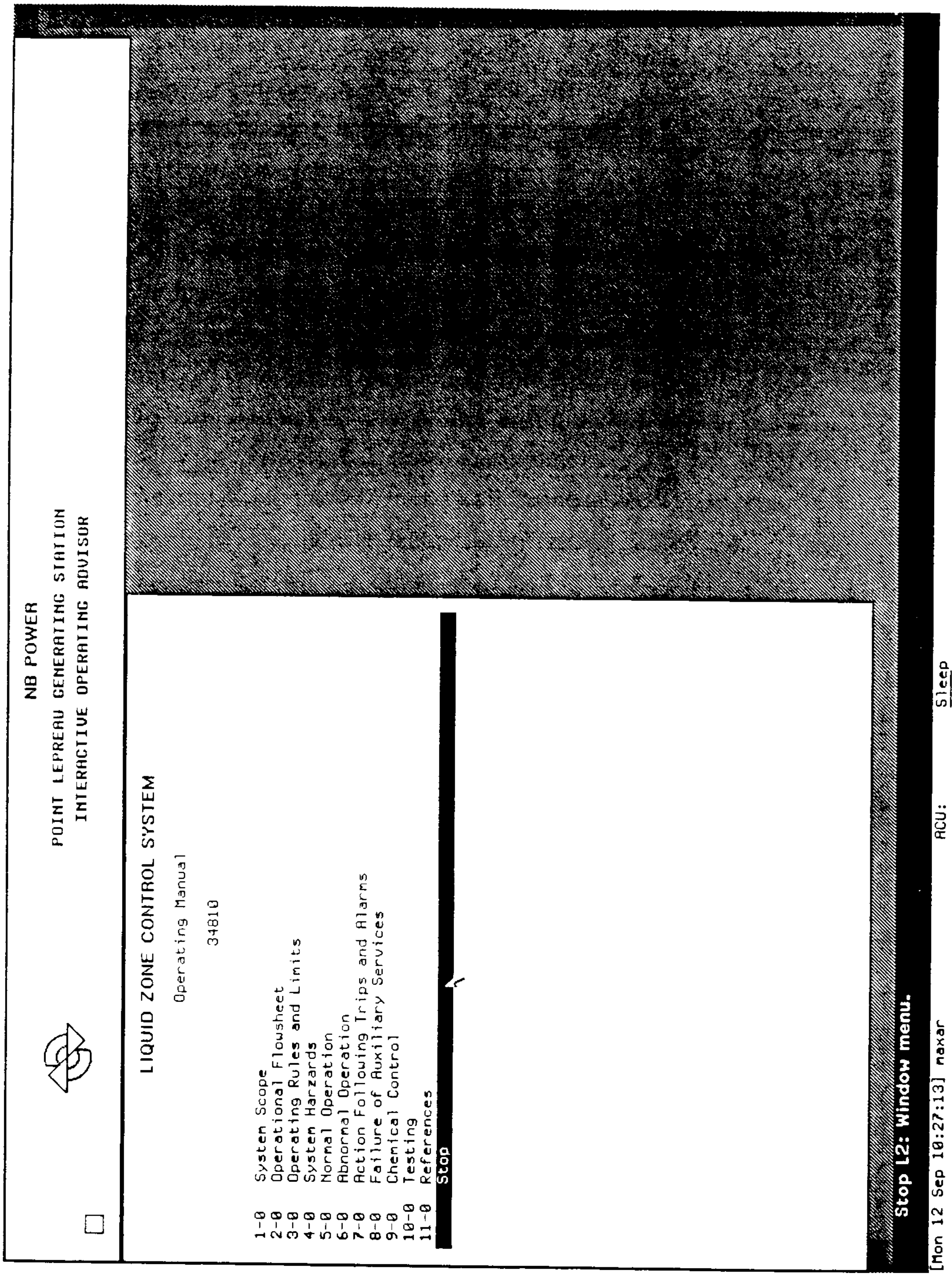


Figure 5-1 Main menu for the Liquid Zone Control System.

The Interactive Operating Advisor must provide an intelligent and user friendly interface for the system operator. The power plant system operator's interface to the expert system is through a menu which resembles the standard index found in the system operating manuals. Figure 5-1 illustrates the main menu for the Liquid Zone Control System. This figure is a screen print taken from the prototype expert system running on the Symbolics Lisp machine. The system operator selects a section of the operating manual by pointing and clicking with a mouse or by moving the highlighted bar to the desired menu option and hitting the enter key. The system operator navigates through the operating manual by making menu selections. The right hand side of the screen, which is black in this figure, is called the working area and it is used only when executing an operational procedure.

Figure 5-2 illustrates a typical screen the system operator might see while executing an operational procedure. The menu displayed on the left hand side of the screen is the menu from which the system operator initiated the operational procedure. It remains on the screen as context information. The working area on the right hand side of the screen is divided into two windows. The top window displays information about the current procedure step being executed. This information is taken verbatim from the operating

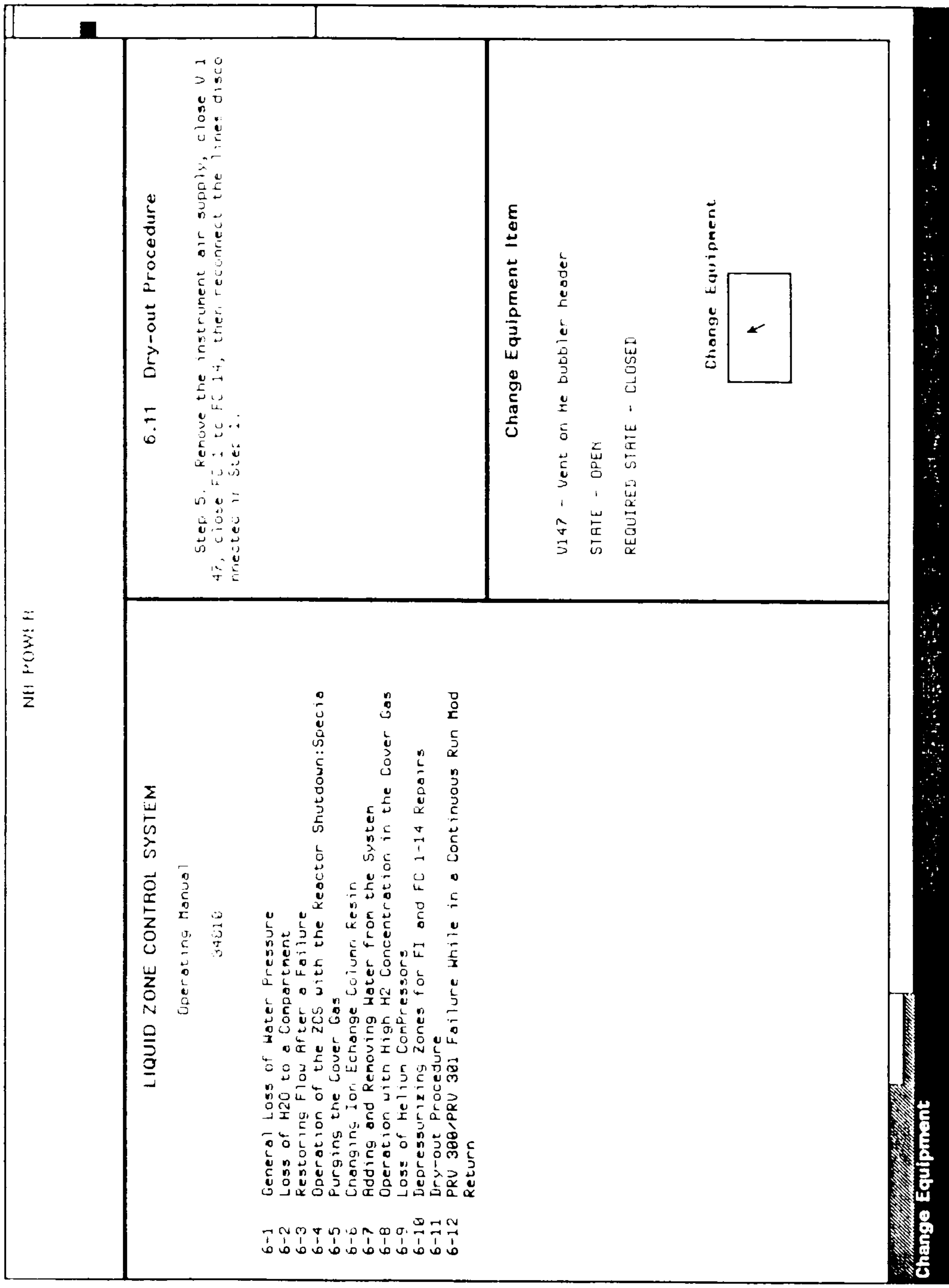


Figure 5-2 Typical Operational Procedure Screen.

manuals. The bottom window is used to interact with the system operator. All messages and equipment state change requests are through this window. Thus, in the top window the expert system shows what it is currently working on, and in the bottom window the expert system shows any deviations and messages to the system operator.

The Interactive Operating Advisor prototype consists of 2 LISP files and 13 ART files. Of the 13 ART files, 3 are more important than the rest. These are the rule file, the equipment file, and the operating manual file. The 10 remaining ART files consists of a menu file and 9 icon files. The 2 LISP files consists of an Interactive Operating Advisor loader file and a functions file.

The Interactive Operating Advisor is started by executing the Interactive Operating Advisor load file. This LISP file performs 3 functions: first, it reshapes the interface windows, second, it issues commands which loads the remaining files, and third, it resets the database so that the system can be run. Figure 5-3 illustrates the Interactive Operating Advisor load file.

The LISP functions file consists of 6 LISP functions which evaluate difficult or complex rule conditions. Implementing the rule groups as described in Section 3.4 requires generic

```
;;; -*- Mode: LISP; Package: ACU ; Syntax: Common-Lisp; Base:10 -*-
; Liquid Zone Control - Load File ; This program loads the files necessary to run
the liquid zone control ; interactive operating advisor.
(reshape-window 'command-window 0 0 570 510)
(reshape-window 'root-menu-window 0 510 570 730)
(clear) ; Clear the database prior to loading
(setq inhibit-fdefine-warnings t) ; Turn off warnings
(load "eval.lisp" :verbose nil) ; Load lisp evaluation functions
(art-load "lzceq.lisp" :silent? t) ; Load the equipment items
(art-load "lzcom.lisp" :silent? t) ; Load the operating manual
(art-load "lzcmn.lisp" :silent? t) ; Load the menus
(art-load "lzcic.lisp" :silent? t) ; Load the equipment icon
(art-load "lzcic1.lisp" :silent? t) ; Load the header
(art-load "lzcic2.lisp" :silent? t) ; Load the proceed icon
(art-load "lzcic3.lisp" :silent? t) ; Load the question icon
(art-load "lzcic4.lisp" :silent? t) ; Load the dry out diagram icon
(art-load "lzcic5.lisp" :silent? t) ; Load the show equipment
(art-load "lzcic6.lisp" :silent? t) ; Load the change equipment
(art-load "lzcic7.lisp" :silent? t) ; Load the profile icon
(art-load "lzcic8.lisp" :silent? t) ; Load the Monitor icon
(art-load "lzc.lisp" :silent? t) ; Load the rules
(reset) ; Reset the database
; (run) ; Run the database
```

Figure 5-3 ART Source Code for Interactive Operating Advisor Load File

rules which can be used in many different situations. Although ART's pattern matching language was very flexible, certain conditions were not easily represented. To resolve this, complex conditions were implemented in LISP. ART's test feature was used which allows LISP functions to be used as part of a rule's conditions.

For example, Figure 5-4 illustrates one of the LISP functions. This function tests one numeric argument against a second numeric argument using one of the six relational

```
(defun my-evall (argl rel-oper arg3)
   (or
      (and(numberp argl)
          (numberp arg3)
          (or
             (and(zl:string= rel-oper "=")(= argl arg3))
             (and(zl:string= rel-oper "<")(< argl arg3))
             (and(zl:string= rel-oper ">")(> argl arg3))
             (and(zl:string= rel-oper "/=")(/= argl arg3))
             (and(zl:string= rel-oper "<=")(<= argl arg3))
             (and(zl:string= rel-oper ">=")(>= argl arg3))))
      (and(not(numberp argl))
          (not(numberp arg3))
          (zl:string= rel-oper "=")(equal argl arg3))))
```

Figure 5-4 ART Source Code for LISP Evaluation Function

operators (=, <, >, /=, <=, >=) or it tests if two character arguments are the same. Using this LISP function, one rule can be used for testing equipment item slots for things such as state = off or pressure >= 150. Without this extension, 7 ART rules, one for each operator with numeric arguments and one for the equal operator with character arguments, would be required.

Both the menu file and the 9 icon files were developed using ARTIST: ART's Interface Synthesis Tool. The artist studio provides tools for creating animated graphics, windows, menus, and asynchronous input/output. The menu file contains the ART code for all the menus found in the Interactive Operating Advisor. Menus are represented using relations and facts. The icons files contain the ART code defining the windows, mouse buttons, and graphics used in the prototype. These are all represented as schemata. The facts, relations, and schemata which define the user interface all get processed directly by ART rules.

The operating manual file contains the ART code which implements the operating manual frame templates as described in Section 3.2. Schemata definitions were developed for the 14 generic tasks. Approximately 500 instances of these schemata were created to represent operational procedures from the Liquid Zone Control System operating manual. This file consists of 43 pages of ART source code representing the Liquid Zone Control System operating manual knowledge. It represents approximately 20% of the operating manual.

All the flexibility ART could provide was needed to implement the operating manual frame templates. For example, Figure 5-5 a) illustrates the schema definition, named requirement, for the operating manual frame template representing the generic task equipment state requirement with "and" logic. The definition of this generic task is a)

```
(defschema requirement
   "defines a system requirement"
   (context)
   (equipment-list))
```

b)

```
(defschema equipment-list
   "slot of requirement"
   (instance-of slot)
   (slot-how-many multiple-values))
```

c)

```
(defschema req-5-1
   (instance-of requirement)
   (context (5-1 2))
   (equipment-list (1 p1 state "=" off) (2 p2 state "=" off) (3 p3 state "=" off) (4 cp1 state "=" off) (
5 cp2  state "=" off) (6 v220 state "=" closed) (7 pv98 state "=" closed) (8 pv106 state "=" closed) (9 p
v187 state "=" closed) (10 pv188 state "=" closed) (11 pv32 state "=" closed) (12 pv33 state "=" closed)
(13 pv2 state "=" closed) (14 pv31 state "=" closed) (15 lcv43 controller "=" closed) (16 lcv44 controlle
r "=" closed) (17 lcv45 controller "=" closed) (18 lcv46 controller "=" closed) (19 lcv47 controller "="
closed) (20 lcv48 controller "=" closed) (21 lcv49 controller "=" closed) (22 lcv50 controller "=" closed
) (23 lcv51 controller "=" closed) (24 lcv52 controller "=" closed) (25 lcv53 controller "=" closed) (26
lcv54 controller "=" closed) (27 lcv55 controller "=" closed) (28 lcv56 controller "=" closed)))
```

Figure 5-5 ART Source Code For a) Requirement Frame Template Definition b) Multi-valued Slot Definition c) Instance of Requirement Frame Template fairly complex (see Section 3.1, generic task 1). ART's schema definition for requirement is quite simple and it consists of only 2 slots. The context slot is used to match the procedure number and step with the control frame. The equipment-list slot is a multi-valued slot of sequences which are the counterpart of a LISP list. Figure 5-5 b) illustrates the ART code which defines the slot equipment-list as multi-valued. Figure 5-5 c) illustrates the use of the requirement schema by showing an instance of the schema being used in procedure 5-1, step 2, of the Liquid Zone Control Operating Manual [Parker and Leclair 1987]. This represents the state the system must be in for shutdown.

The equipment file implements the equipment frame templates as described in Section 3.3. Schemata definitions were created which represent the Point Lepreau reactor, the Liquid Zone Control System, several subsystems, 8 types of equipment, and 105 equipment items. Several relations were defined to make the equipment representation more meaningful.

As previously stated, relations transform the equipment schemata from an equipment list to a more useful and accurate hierarchical representation of the complete physical system. The relations is-a, instance-of, and consists-of with their respective inverse relations kinds, has-instances, and belongs-to were easily implemented in ART. The is-a and instance-of relations with their inverses are standard inheritance relations found within ART. The consists-of relation and it's inverse are user-defined relations. All these relations take the form of a special slot which links one schema definition to another.

Figure 5-6 illustrates ART's representation of the equipment by showing the ART code for 3 schemata definitions. TK1 is

```
(defschema TK1
   "system tank 1"
   (instance-of tank)
   (belongs-to  helium-cover-gas-system)
   (equip-desc "Helium storage Tank")
   (pressure 200)
   (state open))


(defschema tank
   "system tank"
   (is-a equipment)
   (pressure " "))


(defschema equipment
   "generic equipment parent frame"
   (operation-mode manual)
   (equip-desc " ")
   (state " ")
   (location " "))
```

Figure 5-6 ART Source Code for Equipment Schemata Definitions a helium storage tank which belongs-to the helium cover gas system. TK1 is also an instance-of the general schema tank which describes all tanks. Since the instance-of relation is an inheritance relation, TK1 will inherit all the characteristics of the schema definition for tank. Furthermore, since tank is related with the is-a relation to the general schema called equipment, TK1 will also inherit through transitivity all the characteristics of the schema definition for equipment. ART is capable of showing these relations in graphic form such as in Figure 3-21 and Figure 3-22 from Section 3.3 which are screen prints taken from the ART prototype.

The rule file contains the ART rules implementing the rule groups as described in Section 3.4. One rule group was developed for each of the 14 generic tasks. This file contains 90 ART rule definitions and 11 relation definitions.

The relations are used to pass information from rule to rule. For example, a relation named change-equip has been defined to pass the equipment id, slot name, value, and message from the rule which detects an equipment item in an incorrect state to the rule which displays a message to the system operator requesting the situation be corrected.

Of the 90 ART rules that have been developed, approximately 10 rules are used for managerial purposes such as initializing the system and interfacing with the system operator. The remaining 80 rules implement the 14 rule groups which perform the actions described by the generic tasks.

Figure 5-7 illustrates one ART rule, named req-pos1, from the rule group implementing the generic task equipment-state-requirement-with-and-logic. This rule is representative of most of the rules developed for the prototype. The purpose of this rule is to handle the case where an equipment state requirement is already satisfied.

The conditions on the left hand side of this rule require that 3 schemata patterns be matched, that a LISP test be performed, and that several relations not exist in the database. The 3 schemata consist of the following: the control frame (see Section 3.2), an unnamed schema which is an instance of requirement, and the equipment schema referenced by the unnamed requirement schema. The test condition executes a LISP function named my-eval1 with parameters taken from the requirement schema and the equipment schema. The relations indicate that ART is busy doing something else and this rule should wait its turn. This means that the evaluation of equipment states can be

```
; ****************************************************************************
; Requirement - positif 1
; ****************************************************************************

(defrule req-pos1
   (schema control
      (control-pattern (?cxt ?seq))
      (counter ?ct)
      (mode ?md))
   (schema ?sc
      (context (?cxt ?seq))
      (instance-of requirement)
      (equipment-list (?ct ?equip ?slot ?rel-oper ?value)))
   (schema ?equip
      (?slot ?slot-value)
      (equip-desc ?ds))
   (test(my-evall ?slot-value ?rel-oper ?value))
   (not (show-equip ? ? ? ?))
   (not (change-equip ? ? ? ?))
   (not (monitor-equip ? ? ? ?))
   (not (select-equip ? ? ? ?))
=>
   (modify (schema control (counter =(?ct + 1))))
   (if (equal ?md 'yes) then (assert (show-equip ?equip ?slot ?slot-value ?ds))))
```

Figure 5-7 ART Source Code for Requirement Rule

interrupted when partially completed by one of these activities, such as monitor.

There are 2 actions which are performed when this rule is fired. These actions are defined on the right hand side of the rule. The first action is to update the control frame so that the expert system will go on to the next task. The second action is to conditionally assert a fact which requests that the system operator be informed that the state of this equipment item has been verified.

The generic nature of this rule makes it extremely powerful. In the prototype expert system alone, 65 instances of the requirement frame template have been created. Each instance will be processed by this rule. In addition, within each of the 65 instances, the slot equipment-list can be multi-valued. A counter is used to make this rule fire once for each sequence in the slot equipment-list. For example, this rule will fire once for each sequence in the slot equipment-list belonging to the instance of requirement shown in Figure 5-5 c).

5.3 Other Shells

The success of expert systems has resulted in the availability of many commercial expert system building tools or shells. Independent evaluations of these products can be found in the literature [Gevarter 1987, Humphry 1990, Levine 1988]. Almost all expert system development being undertaken today is with the use of an expert system shell [Gartner Group 1989]. Developing an expert system is much easier using an expert system shell than starting from scratch [Grimson and Patil 1987]. In addition, the use of a shell permits the building of an system in an order of magnitude less time than is possible with traditional programming languages [Gevarter 1987].

Each expert system shell has it's own characteristics and the primary objective is to find the shell that fits the problem under consideration. Two critical considerations when evaluating an expert system shell are the way it represents knowledge and the way it searches for solutions. The effective representation of domain knowledge is considered critical to the success of an expert system. The most popular approaches to searching for solutions are forward chaining and backward chaining.

Experience gained in building the two prototypes identified the minimum requirements an expert system shell needs to be considered a candidate for building the Interactive Operating Advisor. The expert system inference engine must support the forward chaining, data driven approach. This is because the Interactive Operating Advisor is not goal driven. It is required to react asynchronously to power plant system events. The rules must provide a powerful pattern matching language and the use of variables. This is necessary to develop generic rules which can be used for all power plant operating manuals. Necessary knowledge representation schemes are facts, relations, and frames. The frames should support multi-valued slots and inheritance. These two advanced features allow frame templates to be developed which represent concisely the operating manual knowledge. A good user-interface and the ability to handle large knowledge-bases are necessary. These requirements limit the choices to high-end expert system shells. Two shells, in addition to ART and it's subset ART-IM, which provide the above required features are Goldworks [Levine 1988] and the Knowledge Engineering Environment (KEE) [Martin and Oxman 1988].

It is interesting to note that functional specifications for a common set of tools facilitating the application of AI to the nuclear power industry have been developed [EPRI NP-4141 1985]. In this study, the Electric Power Research Institute (EPRI) has identified several extensions to the basic AI system that would be helpful in developing effective expert systems in the nuclear power industry. The study used the KEE expert system shell as a basis for analyzing 12 AI nuclear power applications. Extensions to the KEE expert system shell were identified to meet the requirements of the 12 applications. These extensions are: rule building facility, plant modelling facility, simulation facility, data base interface, math modelling facility, machine-machine interface, alarms processor, and real-time processor.

5.4 Prototype Evaluation

The most accepted methodology for building expert systems is through prototyping and incremental development. An expert system, such as the Interactive Operating Advisor, has a long period of immaturity during which it is developed, tested, and incrementally refined [Liebowitz and De Salvo 1989]. The evaluation of the expert system continues throughout the entire lifetime of the expert system. At some point, the expert system passes the prototyping stage and is certified field ready. There are no accepted general metrics for evaluating an expert system, although research is being carried out in this area [Liebowitz 1986].

The prototype expert system, Interactive Operating Advisor, was demonstrated on August 24, 1988 to Point Lepreau staff and to Atomic Energy of Canada Limited (AECL). Both parties were interested with the work that had been done and indicated that the project was progressing in the right direction. The demonstration resulted in a short list of change requests and in plans to demonstrate the system to additional staff from Point Lepreau.

The following list summarizes the proposed change requests.

1. The menu size should be reduced and the working area size increased in order to better utilize the space available on the screen (see Figure 5-2).

2. The system should display multiple procedure steps in the working area.

3. The system operator should have the capability of paging forward and backward through the steps of a procedure.

4. The Interactive Operating Advisor should have the capability of printing field work orders when a procedure step requires work be performed in the plant.

5. A faster access to actions following trips and alarms should be established.

6. Additional operating manuals such as the Emergency Operating Procedures (EOP) should be prototyped for demonstration purposes.

Chapter 6 AUTOMATED KNOWLEDGE ACQUISITION

The Interactive Operating Advisor Architecture has a generic component and a specific component (see Figure 3-2). The generic component, which is comprised of rule groups and frame templates, is for all operating manuals. The specific component, which is comprised of menus and instances of frame templates, is developed for each operating manual to be represented in the expert system.

The operating manual knowledge is the keystone of the Interactive Operating Advisor. This section suggests two ways of automating the transfer of the knowledge contained in the operating manuals to the knowledge representation structure of the expert system.

The most difficult task in building an expert system is the problem domain knowledge acquisition. This task is often the most time consuming and costly. Often, it is a bottleneck in the development of an expert system [Hayes-Roth, Waterman, and Lenat 1983]. Many systems have been developed to aid in the knowledge acquisition process. Some of the most renowned are Teiresias [Buchanan and Shortliffe 1985] and Knowledge Acquisition System (KAS) [Martin and Oxman 1988]. Some experts believe that knowledge acquisition tools will be the most important contribution to the furtherance of expert systems in the commercial world [Martin and Oxman 1988].

A significant amount of research is being done in the knowledge acquisition area. Many different approaches, such as automated learning, representation languages, intelligent editors, and acquiring knowledge directly from text books are being considered. It has been recognized that in special cases, the transfer and transformation of knowledge can be automated or partially automated [Hayes-Roth, Waterman, and Lenat 1983]. It has also been suggested that problem domain knowledge may be extracted from some other automated source such as a database [Martin and Oxman 1988].

A complication to be avoided in knowledge acquisition is representation mismatch. The closer the expert system knowledge representation is to the expert's representation, in this case the operating manuals, the easier it is to determine if the knowledge is correct. The Interactive Operating Advisor uses a semantic network of frames to represent the operating manuals. Semantic networks provide a natural representation of the operating manuals. The hierarchical nature of the operating manuals, chapters, sections, procedures, and procedure steps, and their inter-relationships are easily represented. It is believed that this natural representation of the operating manuals will aid in the knowledge acquisition process.

Two factors make the Interactive Operating Advisor a candidate for partially automated knowledge acquisition. First, the problem domain knowledge is completely contained in the operating manuals which adhere to a standard format and structure. Second, the expert system's representation is very close to the operating manual structure. Thus, the knowledge is readily available and the expert system's representation does not require extensive modifications.

The first approach to automating knowledge acquisition takes advantage of the highly structured expert system knowledge-base. This knowledge-base consists of a semantic network of instances of frame templates. The approach consists of building a front end system which would serve as an intelligent editing program. The editing program would provide tools to define the nodes and links of the semantic network representing an operating manual. As explained in Chapter 3, the nodes and links are instances of pre-defined frame templates. The editing program knows about these frame templates and provides an intelligent interface. In this case, knowledge acquisition becomes a process of selecting frame templates and filling the slots.

A second approach is possible because the operating manuals have been entered into a word processing system (Word Perfect 5.0). It is possible to export the operating manuals as ASCII text from the word processor. Programs could be written to scan this text and output a series of tokens. These tokens could then be processed to extract and re-format the knowledge into the expert system's knowledge representation scheme. For example, a search for keywords such as chapter, section, procedure, and procedure step, could be used to build a tentative operating manual structure. A search for equipment items could build the equipment file. Blocks of text where no processing is required could be imported directly into the expert system. This approach would aid the knowledge acquisition process by reducing the amount of data entry, but is not complete and the results must be verified.

Chapter 7 DISCUSSION AND CONCLUSIONS

7.1 General

Power plant system operating manuals contain the accumulated wisdom of many people devoted to the safe and efficient operation of power plants. The knowledge contained within these operating manuals, in its current paper form, is not quickly accessible by the system operators. The application of expert system technology, as discussed in this thesis, transforms the passive paper manuals into dynamic accessible knowledge.

Some experts believe that the Three Mile Island and Chenobyl nuclear power plant disasters can be directly attributed to operator error. The Interactive Operating Advisor could improve power plant safety and reliability by making the operating manual knowledge more readily accessible. Based on this knowledge and the current state of the system, the Interactive Operating Advisor can advise the system operator on the appropriate course of action during system start-up, running, and shutdown, under both normal and abnormal operating conditions. The Interactive Operating Advisor expert system does not suffer from fatigue, tunnel vision, or other human ailments.

An expert system extends the functionality of the operating manual knowledge far beyond that of the paper manuals. First, the expert system can serve as a training tool for novice system operators. Second, the operating manual knowledge-base can be augmented with the knowledge of the best power plant system operators, making this knowledge available to all the system operators. Third, the expert system can aid in the verification of the correctness and completeness of the operating manual knowledge. For example, after the shutdown procedure is performed, does the system pass the shutdown state verification? Under normal operating conditions, are all the equipment items in their correct state as indicated in the operating manual appendix?

The approach to building the expert system described in this thesis is unique. Instead of using rules and frames directly to represent the operating manual knowledge, building blocks of different types of reasoning, called generic tasks, have been identified. Each building block is generic and widely useful as a component of a more complex reasoning task. This provides the developer with a very high level tool which encodes both the problem solving strategy and the knowledge representation associated with the power plant system operating manuals. The necessity of such a higher level approach to developing expert systems has been recognized in the literature [Chandrasekaren 1986].

The Interactive Operating Advisor does not fall neatly within one of the standard categories used to classify expert systems. The system aids in applying the knowledge contained in well structured operating manuals with significant complexity which occur in power plants and process plants. Characteristics of control, training, knowledge fusing, and monitoring categories can be found in the Interactive Operating Advisor.

7.2 Future Directions

The complex nature of power plants coupled with the growing concern about their safe and efficient operation implies a significant amount research will be carried out in this area in the future. This thesis investigated the application of expert system technology when applied to a power plant operating manual. Although interest was expressed in this project by Point Lepreau, implementing such a system for production use requires more research and development. For example, does the representation of the 10 primary operating manuals result in so many facts and frame instances that the expert system performance degrades significantly?

Point Lepreau has been working on a gateway to their plant control computer. The target finish date for this project is the fall of 1990. This gateway will provide access to the plant system parameters in real time. The next logical step for the Interactive Operating Advisor is to connect to this gateway and re-evaluate the system's performance using real time parameters.

Further work is required in the area of automating the power plant operating manual knowledge acquisition. Chapter 6 suggested two approaches to automating the knowledge acquisition process. It is believed that the second approach which transforms the Word Perfect 5.0 files containing the operating manuals into the form recognized by the expert system is the most beneficial. This approach should be investigated and used to add the knowledge from several operating manuals to the Interactive Operating Advisor knowledge-base.

Once the link to the gateway has been established and the knowledge from several operating manuals has been represented in the expert system, the Interactive Operating Advisor should undergo a trial period of testing and evaluation. The performance of the expert system, in terms of consistency, correctness, effectiveness, and speed, needs to be formally evaluated.

References

B.G. Buchanan and E.H. Shortliffe. Rule-Based Expert Systems. Reading, MA: Addison-Wesley, 1985.

B. Chandrasekaren. "Generic Tasks in Knowledge-based Reasoning: High Level Building Blocks for Expert System Design." IEEE Expert Vol 1, 1986, pp 23-30.

B.D. Clayton. "ART Programming Tutorial - Volume 1". Inference Corporation, Los Angeles, CA, 1987.

W.F. Clocksin and C.S. Mellish. Programming in Prolog. Second Edition, New York: Springer-Verlag, 1984.

P.T. Dean. "Feedwater Heater and Separator Drains". Point Lepreau, NB: Point Lepreau Generating Station Operating Manual 43130, September 1986.

EPRI EL-4323. "Artificial Intelligence Technologies for Power System Operations." Pittsburg, Pennsylvania: Carnegie-Mellon Univ., 1986.

EPRI NP-4141. "Functional Specifications for AI Software Tools for Electric Power Applications." Menlo Park, CA: Intellicorp, 1985.

L.D. Erman et al. "Hearsay II speech-understanding system: Integrating knowledge to resolve uncertainty." Computing Surveys, Vol. 12, No 2, 1980, pp 213-253.

C.L. Forgy. "On the Efficient Implementation of Production Systems." PHD Thesis, Carnegie-Mellon Univ., Dept of Comp. Sci., 1979.

C.L. Forgy. "OPS5 User's Manual." Technical Report CMU-CS-81-135, Carnegie-Mellon Univ., Dept of Comp. Sci., July, 1981.

Gartner Group. "Strategies and Disciplines for Software Engineering." Stamford, CT: Gartner Group Inc., Feb 1989.

W.B. Gevarter. "The nature and evaluation of commercial expert system building tools." IEEE Computer, May 1987, pp 24-41.

W.E.L. Grimson and R.S. Patil. AI in the 1980s and Beyond. Cambridge, MA: MIT Press, 1987.

P. Harmon and D. King. Expert Systems - Artificial Intelligence in Business. New York: Wiley Press, 1985.

F. Hayes-Roth. "The knowledge-based expert system: a tutorial." IEEE Computer, September 1984, pp 11-28.

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

S. Humphry et al. "AI Tools for PCs Tackle App Development." PC Week, February 1990, pp 86-89.

A.R. Johnson. "How to write a power plant operating manual". Point Lepreau, NB: RD-01364-P2 Rev 0, 1988.

A.R. Johnson, B. K. Patterson, and W. Pilkington. "Man-Machine Interface Improvements for Emergency Operating Procedures at Point Lepreau". Point Lepreau, NB: 1988.

K. Levine. "The Age of GoldWorks." PC Tech Journal, May 1988, pp 68-81.

R. I. Levine, D. E. Drang, and B. Edelson. AI and Expert Systems. New York: McGraw-Hill, 1986.

J. Liebowitz. "Useful Approach for Evaluating Expert Systems." Expert Systems, Vol 3, 1986, pp 86-95.

J. Liebowitz and D.A. De Salvo. Structuring Expert Systems. Englewood Cliffs, NJ: Yourdon Press, 1989.

R.K. Lindsay et al. Applications of Artificial Intelligence for Organic Chemistry: The Dendral Project. New York: McGraw-Hill, 1980.

G. Luger and W. Stubblefield. Artificial Intelligence and the design of Expert Systems. Redwood City, CA: Benjamin/Cummins Pubishing, 1989.

L.R. Lupton, L.L. Anderson, and R.A.J. Basso. "The AECL Operator Companion - An Overview." Chalk River, Ontario: Atomic Energy of Canada Limited, 1989.

J. Martin and S. Oxman. Building Expert Systems - A Tutorial. Englewood Cliffs, NJ: Prentice Hall, 1988.

W.A. Martin and R.J. Fateman. "The MacSyma system." Proc. Second Symp. Symbolic and Algebraic Manipulation, 1971, pp 59-75.

J. McDermott. "R1: An expert in the computer systems domain." Proc. First Annual Nat. Conf. Artificial Intelligence, 1980, pp 269-271.

M. Minsky. "A Framework for Representing Knowledge." The Psychology of Computer Vision, edited by P. Winston, pp. 211-277. New-York: McGraw-Hill, 1975.

N.J. Nilson. Principles of Artificial Intelligence. Palo Alto, CA: Tioga, 1980.

D. Parker. "System training manual - liquid zone control". Point Lepreau, NB: Point Lepreau Generating Station System Training Manual TR(A) 34810 Rev 0, June 1980.

D. Parker and S. Leclair. "Liquid zone control". Point Lepreau, NB: Point Lepreau Generating Station Operating Manual 34810, December 1987.

B. Patterson. "Personal Communication" 1989.

W. Siler and D. Tucker. FLOPS - A Fuzzy Logic Production System: User's Manual. Birmingham, AL: Kemp Carraway Heart Institute, 1986.

E.H. Shortliffe. Computer-based Medical Consultation: Mycin. New York: American Elsevier, 1976.

H. Storey, A.R. Johnson, B. K. Patterson, and D.L. Stafford. "A Computerized Monitoring System for Emergency Plant Operating Conditions at the Point Lepreau Generating Station". Point Lepreau, NB: 1988.

VITA

Candidate's full name:	Gene Joseph Maillet
Place and date of birth:	Waltham, Massachusetts January 19, 1962
Permanent address:	Comp. D0 Site 20 SS 3 Fredericton N.B. E3B 5W9
Schools Attended:	Clement Cormier High School Bouctouche, N.B. 1976 - 1980
Universities attended:	University of Moncton B.A.A. (Business
Administration)	
	B.I.A. (Computer Science) 1980 - 1986
	University of New Brunswick 1987 - 1990
Publications:	Nil