CLN 4/P. 37

TOWARDS A GENERIC USER SUPPORT SYSTEM (GUS)[1]

Wm.J. Garland
Department of Engineering Physics
McMaster Univ., Hamilton, Ontario, CANADA, L8S 4M1

W.F.S. Poehlman
Department of Computer Science and Systems
McMaster Univ., Hamilton, Ontario, CANADA, L8S 4M1

R.J. Wilson
Engineering and Computing Services
82 Rayne Avenue, Oakville, Ontario, CANADA, L6H 1C2

A. Bokhari
Department of Computer Science and Systems
McMaster Univ., Hamilton, Ontario, CANADA, L8S 4M1

ABSTRACT

One relevant issue confronting the operator of a nuclear power plant is information and task overload. The goal, then, of the various developers is to improve the current situation by creating an environment where the operator can perform at optimum capacity. This is to be accomplished by providing tools and techniques which reduce operator involvement in low level tasks (freeing up time for higher level cognitive tasks) and assist in information and knowledge manipulation so that high level tasks can be performed more efficiently.

This paper reviews the operational environment of nuclear plants from the viewpoint of engineering design and from the viewpoint of the operators and technical support staff. Principles are advanced for a generic operator companion. These design principles are being applied to a test case: the Pt. Lepreau NGS secondary side central sampling system.

INTRODUCTION

As energy sources of electric power turned nuclear, the required control technology increased in complexity. Through necessity, straightforward analog systems yielded to the more capable digital circuitry. This allowed improved measurement functionality and remote telemetry. What had originally been, in the first generation, a one-sensor-one indicator philosophy, rapidly grew, forcing the operator into information overload, particularly when the process came under stress conditions. In the second generation, such improvements as attribute-based displays of information and alarm filtering became the norm in an attempt to reduce operator loads. These assistance techniques, however, could not compensate for further increases in sensor density and complexity. Thus, in the third generation, the inclusion of computer-based on-line operator aids is being pursued. Recent findings seem to indicate that, compared to the U.S.A., Canada either leads the way or is rapidly gaining ground in adopting new technology in such areas as technology insertions (retrofit) in existing plants and new enhancements beginning at the design level [BHA91].

[1] Canadian Nuclear Society Fourth International Conference on Simulation Methods in Nuclear Engineering, Montreal, Canada, June 2-4, 1993.

This paper advances the notion of a Generic User Support system (GUS). To be effective, any operational aid, be it a simple display or a fully interactive control system, must be designed with the operational environment in mind. The operational environment is comprised of the installed systems of the physical plant (including hardware and software controls) and the human operators and technical staff. Both the plant and the humans have a number of key characteristics that dictate the shape of the tools used in that environment. Consequently, a number of attributes are delineated herein, leading to the design of a generic user support system. A specific instance of the GUS design underway at McMaster is introduced accordingly.

THE PROBLEM DOMAIN

First we focus on the physical plant. One of the most obvious features is its distributed architecture. Plant operations are diverse and multifaceted. Events can happen anywhere in the system. The breadth and depth of plant operations required that the plant be ENGINEERED by decomposition into layered sub-systems (functional abstraction, otherwise known as piece-wise refinement). In operations, operator overload, response time, etc., necessitate procedural responses. The operator is never left with an open ended question. For instance, if the plant state is unknown, there is a definite procedure to follow. At any given moment, there are a limited number of alternatives to choose from. Successful operating plants are, by definition, procedural and pre-enumerated.

Events and reactions to events can occur asynchronously; that is, some sub-systems have no requirement to adhere to the operational milestones of other sub-systems.

Although the nuclear plants have been engineered to be bound by procedures and defined choices, these plants are complex and typically the procedures are often limited to what should be done given that a choice has been made. Choosing between alternatives is quite another matter. The hallmark of a complex system is the existence of higher level functioning or reasoning to deal appropriately with this complexity. This higher level functioning is often termed INFERENCING. For today's plants, much of the required heuristics are available only from human operators and this is not likely to change in the near future.

In the diverse environment of the nuclear plant, the consultation of many disparate knowledge bases involving the many plant components and plant views is required for problem solving. For example, Darlington NGS has an Equipment Monitoring System and Pt. Lepreau has a Chemistry Monitoring System. The various agents (machine and human) will operate (inference) on these knowledge bases in diverse ways.

Expertise from many disciplines (reactor physics, chemistry, thermalhydraulics, control, safety, etc.) are brought to bear in the problem solving process. At the moment, only the human operator is capable of multidisciplinary integration and this will not change in the foreseeable future.

Different mental models are used even within a discipline. This refers to the different problem solving strategies of the technician / operator vs. the engineer / designer. This is elaborated in the section on cognitive issues.

The overall control strategy used at the plant is of central importance to the GUS design. For the most part, regulations require that the human remain 'in the loop', ie, in control. The operator has at his or her disposal many tools to enhance performance but the operator is very much in the driver's seat. This is so not just because regulations require it. It is so also because operators, like engineers do not wish to divest their authority to a machine. Machines are tools, no matter how intelligent they appear to be. This issue has been referred to as the machine-centred approach vs. the human-centred approach.

Nuclear power plants operate, of course, in real time and require real time control. 'Real-time' can only be defined in terms of the information in question, ie., real-time pressure data probably requires updating every second or so, whereas refuelling history data requires updating only 20 or 30 times a day. In short, real events happen when they happen and an appropriate and timely response is required.

To respond to real events in a complex plant, analysis is required. Much of the control is automatic but a significant role is played by the operators who must reason about the situation at hand. Figure 1 illustrates the roles played by the human operator with respect to the plant and plant control (as portrayed in more detail by [LUP90]). It is not just a question of being fast. Rather, bounding the solution time is the issue. Can a sufficiently good solution be found within the time required? The goal here is to 'sufficify' rather than 'optimize'. The successful operator or control system reaches the appropriate conclusion and implements it before the real system (the plant) does. Approximate solutions can be refined later. Here again, the heuristics are not well delineated.

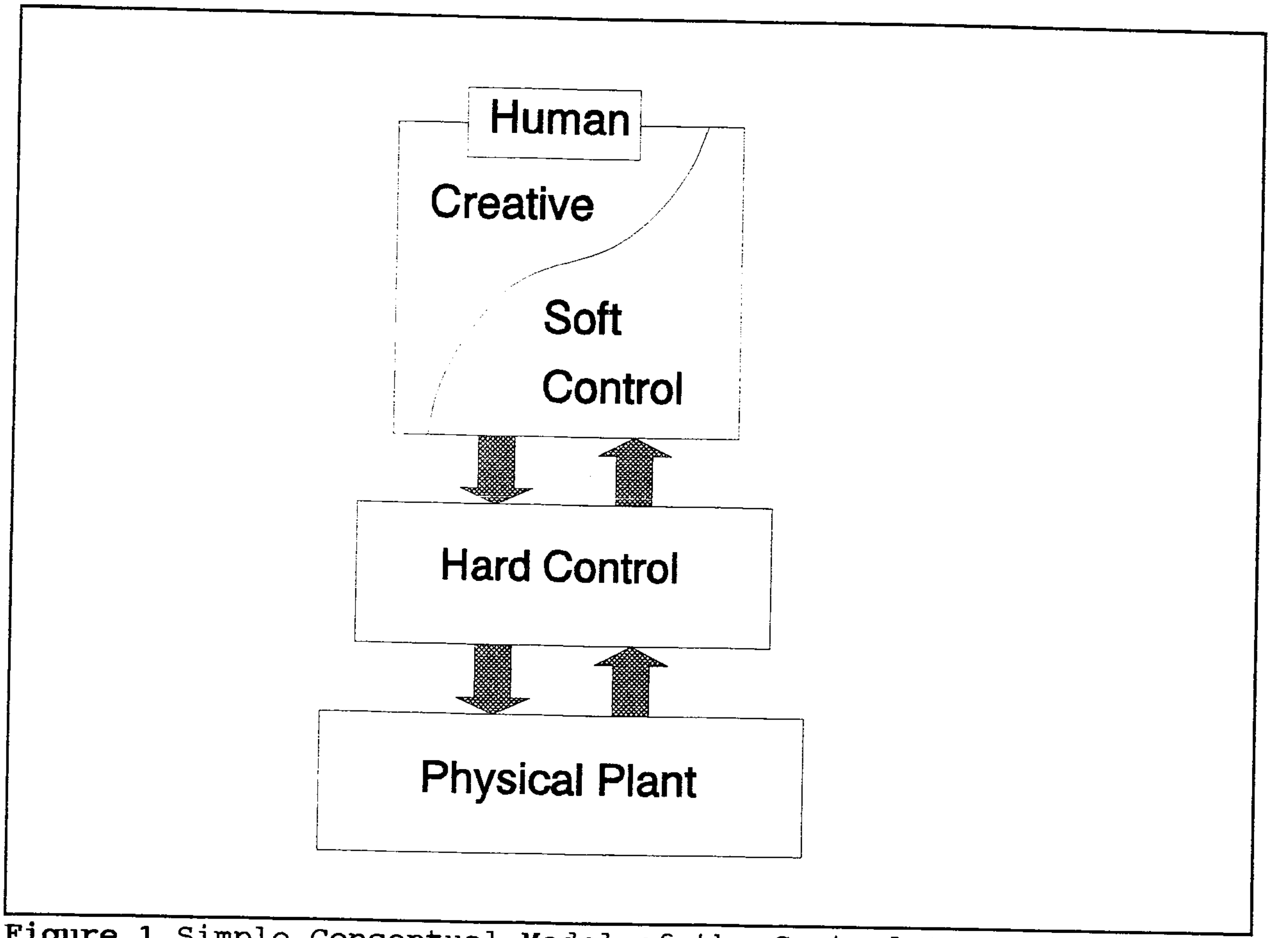

Figure 1 Simple Conceptual Model of the Control Process

The sub-systems of distributed architectures are typically hierarchically organized. Compared to higher level sub-systems, lower level sub-systems respond more quickly to more basic inputs in a more procedural manner. An example would be a standard proportional-integral-differential (PID) controller. Typically data flows upward through the hierarchy and at each level is transformed in some way. Higher level systems need not be informed about un-necessary information. This is information hiding. Higher level subsystems (the operator constitutes the highest level) involve more highly abstracted data which requires higher levels of cognitive analysis and which, by their very nature, are performed more slowly than lower level responses. This is temporal abstraction. Figure 2 illustrates the functional and temporal abstraction typically found in a complex plant.

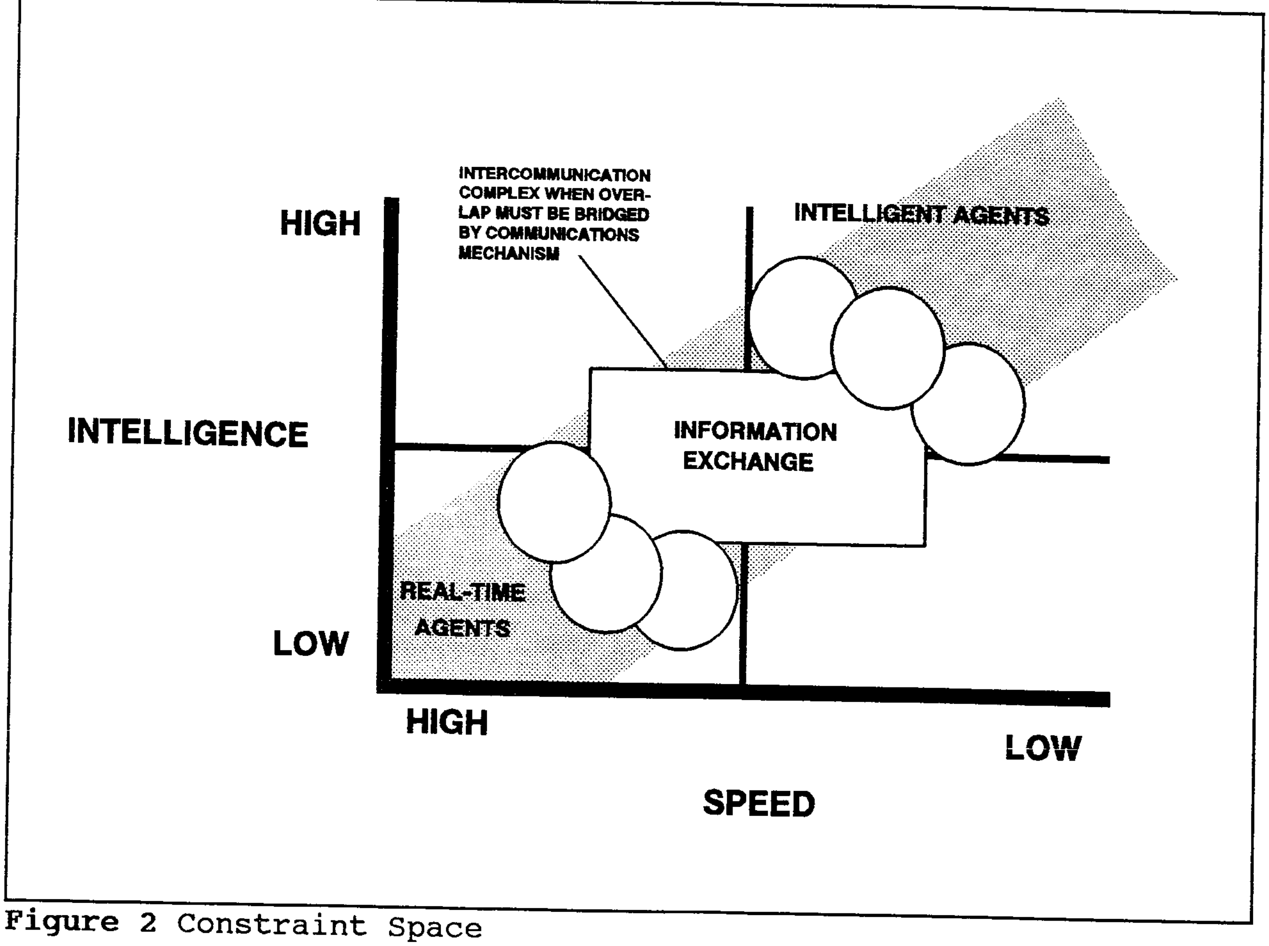

Figure 2 Constraint Space

COGNITIVE ISSUES: Problem Solving Strategies

The mental model of the designer or engineer as developed in Rasmussen's book [RAS86] is one that is based on functional decomposition. The engineer poses: How does the plant work? What is broken? What measurements must be taken? What is the functional decomposition of the plant? How do the parts interact? How can one simulate it? This 'mechanology' led to alarm based annunciation, control room displays and controls grouped by system (functional decomposition), sensoritus, and information glut without enhanced knowledge. The view of the plant taken was that of the design engineer - this is how and why it works - here are all the details, etc. Ergonomics was commonplace but limited to 'knobology'. Make the knob bigger, use a red light here, etc. This kind of thinking leads to products like the VCR - machines with attractive lines, buttons that 'feel' right, on-screen programming and 64 button remote controls - full featured functionality from the comfort of your armchair. And unusable for anyone except a technoid! Problem solving strategies here relies on a deeper than average understanding and use of specific knowledge. Contrast this to the more typical user.

The user, it is now recognized, is the operator, not the design engineer. The operator's mental model of the plant is closer to that of a technician - the plant is a collection of many systems, most of which are treated with generic algorithms for fault diagnosis and treatment of event symptoms, irrespective of the system in question. Rasmussen's figure (reproduced in part in Figure 3) illustrates this generic

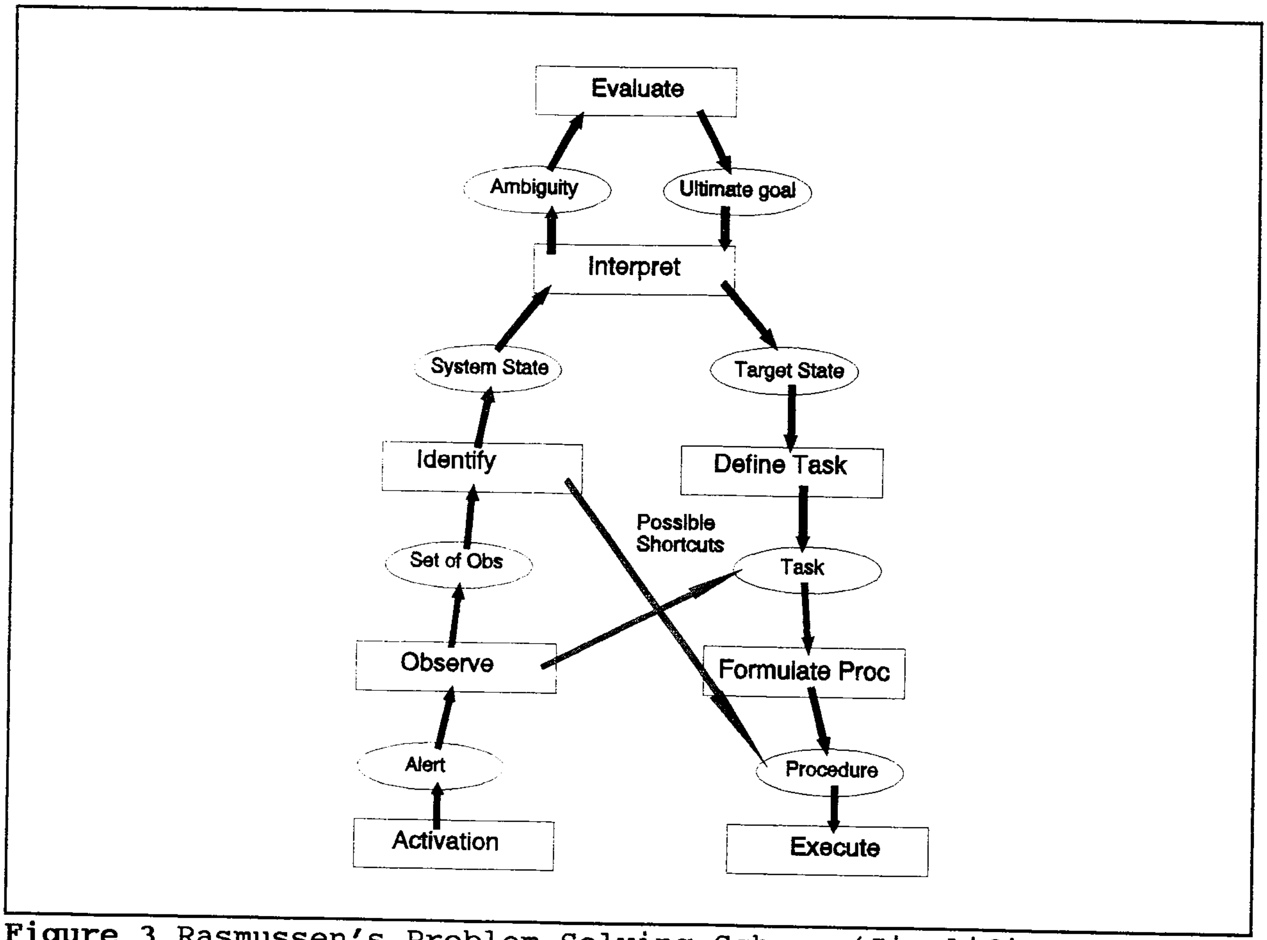

Figure 3 Rasmussen's Problem Solving Schema (Simplified).

algorithm. The generic algorithm for problem solving is to observe and identify the state of the situation, interpret, evaluate, plan actions and execute the actions. Rasmussen notes that shortcuts can be taken at any stage. In fact, most of what we do involves shortcuts to some degree. ALL problem solving is covered by this figure but the technician often employs strategies and tactics that do not rely heavily on a detailed knowledge of system and component behaviour. That is, short cuts to Rasmussen's full solution path are taken. This is a form of shallow reasoning and is good most of the time. This is not to say that the operator does not have a detailed knowledge of the systems and components. He or she indeed does. It is simply that the problem solving scheme is not as system dependent as for the engineer.

Another point of note is the fact that expertise is composed of both knowledge and the ability to manipulate that knowledge. It has been observed and is widely recognized in the literature that it is the vast knowledge-base that exemplifies the expert rather than raw inferencing capability. Take an expert and place him or her in novel territory and you get decidedly non-expert behaviour. Take a genius and place him or her in the expert's territory and you do not get expert behaviour. The reactor environment is a case in point; operator actions require mostly procedural knowledge and moderate inferencing capability. The knowledge-base (composed of facts and heuristics) is then very important. The knowledge-base is necessarily system or component specific and it is unlikely that a general knowledge-base can ever be conceived. This leads to the paradigm of message passing via an electronic blackboard or some other mail handling system as a means of allowing disparate agents (models or codes or humans, etc.) to interact. Knowledge-base design is, then, more about the design of the format and content of the messages being processed. It is at this level that one is concerned about how the operator or engineer interacts with the system being controlled. The result of this is that the objects being manipulated (the knowledge base) need to be defined before they can be manipulated (by the inferencing or procedural engine). But how we define these objects will depend on the mental model chosen. This leads to a dilemma: the system needs to be functionally decomposed along the lines of the physical or engineer's mental model, whereas, this is an inappropriate model for the operator. The plant is organized along the same hierarchial lines as the design engineer's mental model. That organization expects the human to provide the top level knowledge based control. The operator, however, spends much of the time at the rule based and skill based levels. Figure 4 illustrates this point. The issue is not a trivial one and, as such, deserves careful consideration in the design of operational support systems. No-one to date has demonstrated a schema to solve this dilemma. This mental model mismatch plus a machine-centred bias are arguably the leading causes of the failure of artificial intelligence based support systems.

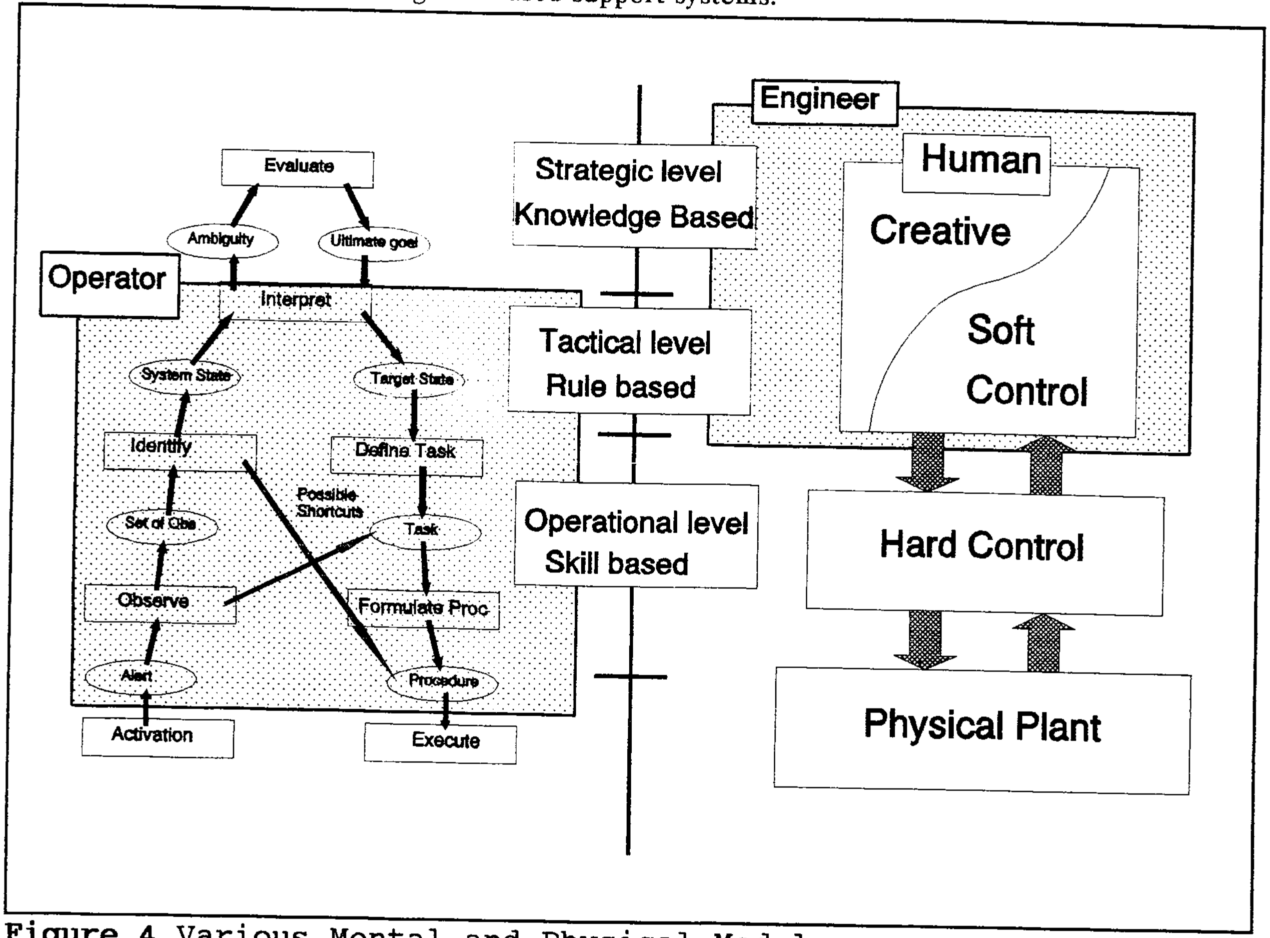

Figure 4 Various Mental and Physical Models.

GENERIC USER SUPPORT SYSTEM DESIGN

From our study of the works of others and from our own investigations, we have found that there are some general principles that apply to the design of operational support tools. For the most part, these stem directly from traditional engineering design practices. Customary design features include flexibility, modularity, incremental growth capability, and independence of modules from the control structures. We follow these features as much as possible. However, the nature of the nuclear plant environment guides the design and development of useful operator aids. Next we discuss how the characteristics outlined above manifest themselves in the GUS design in particular.

What single aspect is the most important to capture and hence most influences the design of an operator aid? The distributed architecture stands out as central since this dictates the whole character of the station. It IS, in fact, the physical station. Diverse and disparate knowledge bases and asynchronous activity naturally follow. The plant has been ENGINEERED from the ground up and any aid must acknowledge this fact. This complex system must be controlled and this implies measured data, system knowledge and agents that act in this control function. The distributed and diverse nature of the plant also implies that the controlling agents be diverse. No one aid can ever hope to encompass the needs of the operator. Rather, many small aids need to be developed in a coordinated manner following the natural organization of the physical plant and activities of the operational staff. But as soon as one prescribes separate agents in the design of the operator aid, artificial divisional boundaries have been created and these boundaries must be bridged by message passing of some form. This, in turn, implies a message format and a communication medium. Since the aids may be as physically distributed as the plant, some form of local area network is indicated. There will inevitably be copious quantities of data and it is usually prudent to include a data storage agent which can act as a message coordinator or post office. Such an agent is called a blackboard or, perhaps more correctly, a postboard. By a judicious choice of agents and their duties, message passing can be minimized. Indeed, message passing has proven to be a bottleneck in the past, leading to a design principle of making the agents as proactive, persistent and persevering as possible, meaning that they act with the least instructions, will continue to act until told to stop (or their duties are complete), and will try hard to complete assigned tasks even in the face of incomplete or inconsistent data.

The individual aids, operating in parallel, are of considerable utility by themselves but the true power of the paradigm comes when the various individual aids cooperate in problem solving (concurrent engineering).

We note that this is a conceptual organization. Implementation could vary from single processor, single tasking to multiprocessor, multitasking. The implementation could be on one machine or distributed over many machines (tightly or loosely coupled).

The asynchronous nature of the plant (required for the practical design of any complex system) permits the use of asynchronous agents in our operator aid. It is not necessary for the aid to be asynchronous but the implied independence of the agents is a boon to design. Division of labour and piece-wise refinement are the order of the day, both in plant design and in operator aid design.

So far, our design discussion has lead to a society of asynchronous agents performing their jobs in a loosely coupled architecture, communicating via mail, sharing data and other information. We are free to design any individual agent to best suit the needs of the operator in this area. The level of agent intelligence is not preordained. This design was not by chance for we wish to maintain flexibility and modularity as much as possible to permit growth and adaptability of the installed aid. For instance, present operators might favour the installation of on-line documentation now but as experience with (and confidence in) aids grows and as heuristics are uncovered, more elaborate aids might be requested. Plant operation covers many disciplines and concurrent problem solving is common. To build aids to help in this area requires the discovery of the heuristics used by plant personnel. The GUS design permits such agents to be built but this is not a design priority at the moment since the heuristics are not known and since plant personnel would not likely accept such agents at this time even if they could be built. Concurrent problem solving lies much further along the implementation path.

As mentioned, the plant has been engineered and now it must be operated; but the mental models in the two activities are different, leading to a dilemma for the designer of operator aids. The basis for the plant (and plant models) are different from that of the operator. Interaction with the operator must be on the operator's terms. To make the operator aid more understandable to the user, the aid is developed along anthropomorphic lines of manager, supervisor and technician. Further, the information presented to the user must be consistent with the decision making process of the user. For instance, inevitably, the user is faced with making a choice:

'Of the possible problems facing me, which is the one that I should pursue now?'
One possible technique that has proved successful is scorecarding.

Scorecarding (the tallying of how well alternatives measure up for a number of attributes) are useful for reconciling judgements with facts, symbolics with numerics. As discussed in [GAR90], the solution strategy centres around the elimination of alternatives that cannot be used for one reason or another, and the ranking of those that are left by the use of score cards. The specification of the scoring details is a knowledge engineering exercise that is not trivial but it is one that must be done in some form at some point in the design of any operator aid. Scorecards merely give form to the substance; but most importantly, the form is an appropriate one for decision making for all types of users, including operational support staff. User supplied attribute weights are used to generate the weighted sum to give the total scores used to rank the alternatives in the decision to be made. The score card approach provides an good way to perform the ranking since it emulates the expert methodology of weighing the pros and cons of the alternatives. Generic entropy and standard deviation based algorithms [GAR90], originally developed for Decision Support Systems have been considered as possible means of dynamically altering the attribute weights in order to focus the user's attention on the relevant scores for the problem at hand. The standard deviation method proved superior and is now ready to be employed. This has been implemented for heat exchanger selection and the prototype performed very well, meeting or exceeding all expectations in ease of construction, ease of use, speed, and accuracy in emulating the human expert.

This implementation of scorecarding provides a means of dealing with disparate agents but it does not address the issues of real-time, asynchronous events or concurrency (ie multiple agents acting in parallel and sharing information during the solution). However, they are not precluded either. Such a technique clearly allows the user to remain in control since the aid simply presents the relevant information in a manner conducive to decision making, providing an automatic focus on the important issues and letting the user decide which path to actually follow.

As mentioned, the plant is human centred. Past work on Expert System based operator aids has been machine-centred. This is a term used by Bernard [BER92] to describe the overall design philosophy of building tools that put the machine in control, that is, the computer program tells the operator what to do and when to do it. Natural scepticism coupled with the clearly limited expertise exhibited by Expert Systems ensured failure for attempts at building machine-centred operator aids. Contrast this to the human-centred approach wherein the operator is the primary source of intelligence and is in control. In the human-centred approach, the operator uses the computer programs as tools, as powerful extensions of the operator. The computer may monitor and annunciate but it is the operator who pilots the operation, using the tools when necessary and as necessary. The paradigm shift is profound. Bernard notes that the machine-centred approach is now considered inappropriate.

Discussion on the obvious plant characteristic of real-time performance has been postponed until now since, as important a characteristic as it is, it does not play a central role in operator aid design at the level that it is being discussed herein.

Any operator aid of value must contain knowledge of the plant that is up to date and it must reason about that knowledge at an appropriate pace. Thus, there are two time related aspects of a real-time aid. Our solution here is to functionally and temporally abstract the design so that specialized agents can handle the time critical events like data acquisition, filtering, trending, etc. leaving the interpretation and analysis of the data to agents on a higher level. This is precisely how real-time reasoning is handled in the physical plant. Reasoning heuristics provide the methodology for dealing with the diverse and sometimes conflicting implications of plant data and alarms. Real-time for an operator aid means that the aid must be able to provide aid at the human's pace, not the plant's pace. This is so because the plant has been engineered to be so.

A real-time system is one that reaches its conclusion before the real system does. Can the aid keep up? We need a strategy to cope and one possible strategy is to provide a fast but approximate solution with subsequent refinement later if possible. This is somewhat like the navigational technique of establishing an approximate heading early on and correcting en route.

Reasoning in real-time about real-time data is in its infancy. No one has yet developed truly real-time inferencing and current expert systems are limited to treating time evolution as a series of finite states. Consequently, current operator aid designs should concern themselves with aiding the operator in making real-time decisions by providing the right information at the right time and place in a manner that integrates well with the thought processes of the user. However, nothing in our system precludes the use of operator heuristics (once they are uncovered).

The ideal environment would include a communication highway (like ethernet) to allow peer to peer interaction, involve distributed processing, permit short circuit heuristics (as per Rasmussen), have no implied or enforced control structure (orthogonal to the control plane) and permit parallel and concurrent problem solving. Figure 5 illustrates an implementation that has the required generality and flexibility. No specific control structure is implied; it is simply a fast, inexpensive message passing medium that is available today. Note that incoming plant data is broadcast to all agents but the primary direct users of the plant data are the data acquisition agents and the blackboard. All agents are free to interact with other agents. (the blackboard and the data acquisition agents are just specific instances of agents).

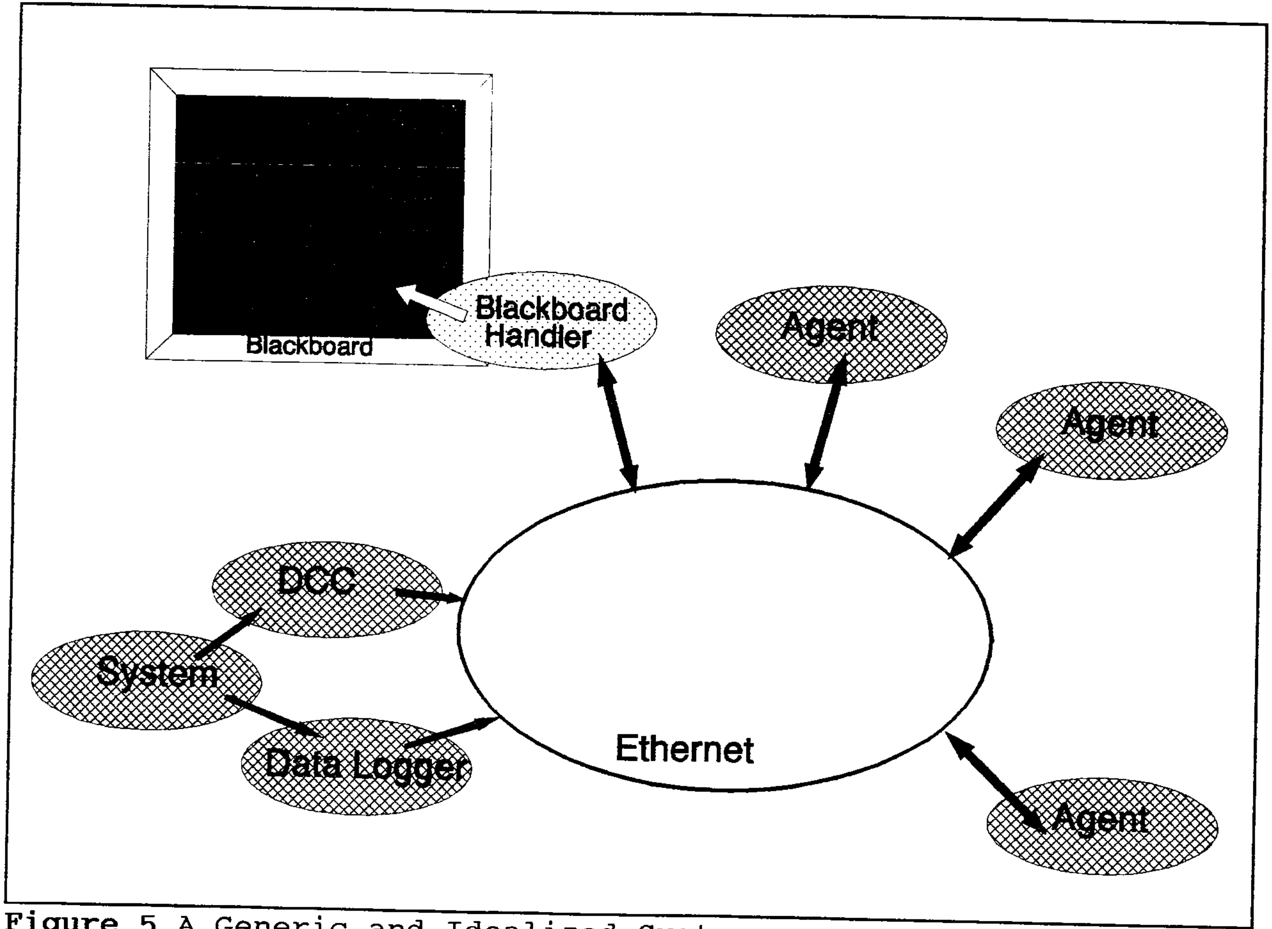

Figure 5 A Generic and Idealized System.

A SPECIFIC GUS IMPLEMENTATION: OPUS

On the suggestion of personnel at Pt. Lepreau, McMaster University is developing, for the Pt. Lepreau NGS Central Sampling System, an advisor on turbine condenser tube leaks and reactor derating due to secondary side chemistry problems. Research on this project, dubbed the Operator / User Support Project (OPUS), began informally in 1990 and formally in November, 1991 with the award of a 3 year NSERC Strategic Grant. One of the projects goals is to demonstrate the utility of the anthropomorphic approach of applying the blackboard paradigm partitioned along the lines of manager - supervisor - technician, allowing symbolic - numeric coupling with the inherent efficiency of asynchronous operation in real-time.

To date, a procedural code has been developed to provide a timing benchmark and to validate the logic. A PC-based multitasking blackboard prototype has been developed, tested and benchmarked for a toy problem [MAH92]. A PC-based blackboard version of the central sampling advisor is currently in beta-testing. To explore the migration of the aid to a distributed architecture, a LAN based on ethernet and TCP/IP between a SUN Sparc Station and 3 486 PC's has been installed at McMaster and a socket based message passing library over the LAN has been established for UNIX-UNIX communication. PC-PC and UNIX-PC socket libraries are under development. Interaction with Pt. Lepreau continues. As has been found in many other knowledge based tool development, the biggest bottleneck is discovering the expert's knowledge and organizing that knowledge in a coherent manner. Plant operators do not have a lot of free time to ruminate for the knowledge engineer's benefit and the process of turning inherent expertise into explicit heuristics is not trivial.

The OPUS system is depicted in Figure 6. Note that the structure of the aid follows the generic principles outlined in this paper. Currently it is not tied directly into any plant data but it could easily be linked to Pt. Lepreau's GATEWAY LAN giving access to existing Chemistry Monitoring System data.

CONCLUSIONS

In conclusion, generic operator aid design principles have been delineated. It has been found that the design follows quite directly from a study of the physical plant and the human operators and technicians. The resulting user support system design is very flexible and is proving to be a solid basis for a specific design aid under development for Pt. Lepreau NGS. There appears to be nothing inherently difficult in implementing OPUS once the knowledge format is cast. The difficult part is the knowledge engineering: the experts have limited time to impart their knowledge to the knowledge engineers and even if adequate time were allocated, the expertise often has to be 'discovered'. These are challenges however, not reasons to default to the status quo. Even if no operator aid were ever to be implemented, the knowledge discovery would justify all the effort.

ACKNOWLEDGEMENTS

This work has been made possible through funding from the Atomic Energy Control Board and the Natural Sciences and Engineering Research Council of Canada, Strategic Grant number STR0118177

REFERENCES

BER92 John A. Bernard, "Issues Regarding The Design and Acceptance of Intelligent Support Systems for Reactor Operators", ICHMT 2nd International Forum on Expert Systems and Computer Simulation in Energy, University of Erlangen, 17-20 March 1992.

BHA91 S.C. Bhatt, "Assessment of the Canadian Instrumentation and Control Technology in Nuclear Power Plants", document produced for the National Science Foundation and department of

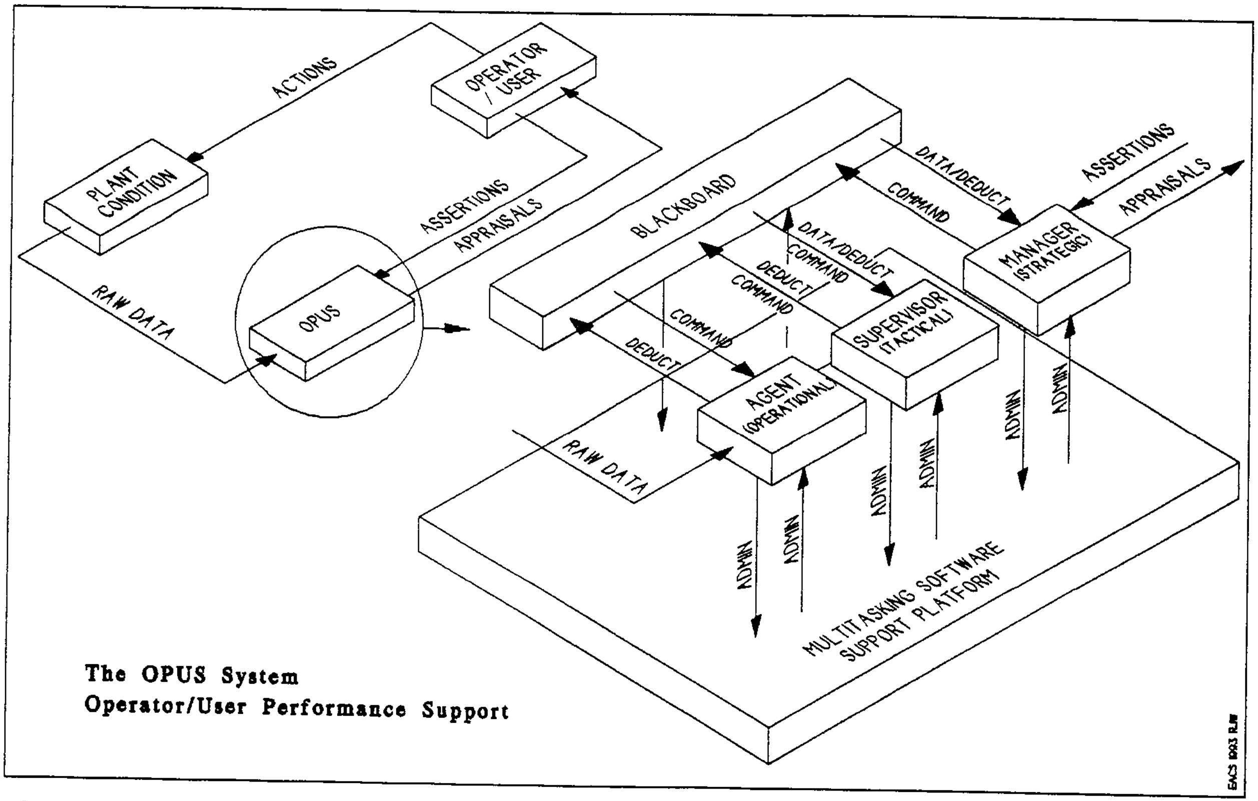

Figure 6 OPUS system overview

Energy, U.S.A., 1991.

GAR90 Wm. J. Garland, "Knowledge Base Design for Heat Exchanger Selection", Engineering Applications of Artificial Intelligence, Vol. 3, # 3, September, 1990.

LUP90 L.R. Lupton, J.J. Lipsett, R.A. Olmstread and E.C. Davey, "Foundation for Allocating Control Functions to Humans and Machines in Future CANDU NPPs", Proceedings of an International Symposium on Balancing Automation and Human Action in Nuclear Power Plants, sponsored by the IAEA and OECD, Munich, July 9-13, 1990, IAEA-SM-315/28 pp 349-367, also as AECL - 10198.

MAH92 A.S. Mahmoud, Wm.J.Garland and W.F.S. Poehlman, "Multitasking Strategies in Support of a Knowledge-based Operator Companion", AIENG'92: Engineering Applications of Artificial Intelligence VII, ed. D.E. Grierson, G. Rzevski and R.A. Adey (Elsevier Applied Science, New York: 1992) pp. 1001-1015, Waterloo, Ontario, Canada, 14-17 July, 1992 (Wessex Computational Mechanics Institute, U.K.)

RAS86 Jens Rasmussen, "Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering", North-Holland Series in System Science and Engineering, 1986, ISBN: 0-444-00987-6.