A FRAMEWORK FOR EXPERT SYSTEMS DEVELOPMENT INTEGRATED TO A SCADA/EMS ENVIRONMENT

V. N. A. L. da Silva¹, R. Linden¹,², G. F. Ribeiro¹

I. INTRODUCTION

UTILITIES have invested heavily in control and supervision equipment, but operation is still dependent on peopleware. All knowledge on the operation and the planning of the power sector are still property of an exclusive group of experts. These experts are capable of making decisions based on logic, heuristics, experience and non-formalized knowledge, such as intuition. Nevertheless, the growth seen in the power companies and the loss of experts, due to retirement, have made utilities understand that it is imperative that they develop new techniques for control, supervision and automation that can preserve corporate intelligence and memory. Hence, it will be possible to use it in the future to train novice specialists so that they can familiarize themselves with situations they have never faced in order to be prepared to solve them faster whenever they may arise.

These needs make us consider techniques that can learn and incorporate previously available knowledge, an area in which Artificial intelligence (AI) excels. AI branches into Fuzzy Logic [1], Genetic Algorithms [2], Neural Networks [3], and, specially, Expert Systems [4], whose main characteristic is to simulate the human expert thought process. Taking those needs in consideration, CEPEL used Expert Systems to develop the project “Expert Systems for Control Centers in Real Time - SAGE–EXPERT”.

Expert Systems are a well proven technology for Energy Management Systems (EMS) applications, having decades of application in this area. They were chosen because their solutions are fully explainable, can be easily documented and incorporate previous knowledge, abilities that neural networks lack. Operators have detailed knowledge of previous faults and there are well documented manuals on the operation under faulty conditions. Even though those bodies of knowledge may not cover all the possible power system faults, they may not be discarded beforehand.

Fuzzy logic was not chosen because alarms have no inherent imprecision, even the numeric ones, which are considered in relation to crisp boundaries. Hence, fuzzy logic’s ability to treat numbers as linguistic concepts would not improve the system’s performance. Genetic algorithms could be considered as a tool to develop rules for the expert systems. Similar approaches, even though directed to fuzzy logic rules, were adopted in several papers for different application areas, such as [5]. This option was not considered in this paper due to the facts that previous knowledge was extensive and an association rule technique had already been used to mine for prospective rules.

In addition to all those features, Expert Systems have a near-human reasoning process and a long successful history in applications installed in Power System Control Centers. Nevertheless, for years now, fewer people have been considering Expert System as a useful tool in EMS because when compared to other AI approaches such as ANN and GA, Expert System suffers from disadvantages, mainly in maintenance. The idea behind this work is also to simplify the maintenance tasks introducing a powerful graphical interface that makes it easier to create and reuse generic topological rules that apply to all power stations that share the same configuration. This way, expert systems can be seen as more competitive when compared to the other AI techniques that have automatic but slow training.

SAGE-EXPERT is a framework that is completely integrated to SAGE, CEPEL’s own SCADA/EMS system. It allows for the creation of new software applications without the development of additional modules for rule processing and communication with SAGE. It is implemented on a standardized programming language and can execute on multiple operating systems without loss of either performance.

¹ CEPEL (Electric Power Research Center), Av. Horácio de Macedo 354, CEP 21941-911, Rio de Janeiro-RJ, Brazil
² FSMA (Salesian College of Macaé), R. Monte Eliseo SN, Visconde de Araujo, CEP 27943-180, Macaé-RJ, Brazil
E-mails:
- V. N. A. L. da Silva – navarro@cepel.br
- R. Linden – rlinden@cepel.br
- G. F. Ribeiro – gfr@cepel.br
or abilities. The software has been tested solely on Linux, Unixware and Sun Solaris operating systems, but, since its programming is completely done in ANSI C language and its structure is simple and modular, portability to any UNIX family environment should be straightforward.

SAGE-EXPERT is a centralized server that can answer to multiple clients simultaneously in a multi-tasking fashion. Its clients can be run in any computer, both local and remote, and communicate with the server using TCP/IP and sockets. The open communication protocol based on XML is simple, text based and can be implemented by any application software, so that any client can use a powerful artificial intelligence tool.

II. EXPERT SYSTEMS

Expert Systems are part of the research field called Artificial Intelligence, whose main goal is to create computer programs that can mimic intelligent behavior [6]. Expert Systems are programs that embed expert knowledge on some specific subject in order to infer new facts, based on currently known facts and rules. Expert Systems are useful in situations where there is a small group of experts that has a high-level of know-how experience and expertise that cannot be easily taught to other workers. Using an Expert System, besides increasing productivity and decreasing the error rate, will allow for formalization of this knowledge and will make it easier for this knowledge transference to occur.

Developing an Expert System means extracting, from persons that can be considered experts, rules, strategies and procedures to solve a pre-determined group of problems, aggregating this knowledge into a computer software system. This process is known as Knowledge Engineering and the person responsible for it (the Knowledge Engineer) must understand Expert Systems as well as have a deep interpersonal ability, profound interviewing skills and an extensive understanding of the human factor impact in order to get the experts to cooperate with the engineering process.

Given the fact that all knowledge is obtained by expert interviews and that this process is difficult and time-consuming, Expert Systems are notoriously narrow in their domain of knowledge. Therefore, they should be considered in domains where the problems to be solved are of the sort that would normally be tackled by a knowledgeable professional using well defined thought mechanisms.

Expert Systems are superior in situations where the number of rules is too high, causing a high response time from a human expert and strong impact on the transferability of this knowledge to new staff members. Besides, Expert Systems will provide a non-emotional response, quicker and more accurate than human experts most of the times.

Creating a new Expert System is a task that does not have severe technical difficulties besides understanding the new programming paradigm, since there are several specialized tools, called shells, that have all the necessary technology embedded in them\(^1\). As stated above, the biggest hurdle to create a new Expert Systems application is the development of the knowledge base, which demands a huge interaction between the Knowledge Engineer and the human experts.

One of the main obstacles for the usage of these Expert Systems, in spite of its high applicability, is the fact that there are no Expert System platforms that are simple, versatile and seamlessly integrated into the Control Center environment.

SAGE is a real time control and supervision software that is widely used in Brazil. Its widely usage makes it an appropriate environment for the integration of such a software, given the fact that it is an open platform. Therefore, it is natural that any software that wishes to integrate to SAGE should use simple and open communication protocols, besides being implemented in an efficient fashion, so that its usage does not cause any relevant impact on the real time software’s performance.

III. FRAMEWORK FOR EXPERT SYSTEMS DEVELOPMENT

A new framework based on Expert Systems was developed at CEPEL in order to serve as a generic tool for knowledge representation. This framework can reside at a central server and serve multiple clients simultaneously, without loss either of capabilities or performance. There are several Expert Systems shells available, but, in order to implement SAGE-EXPERT, it was decided for CLIPS, because it is free and defined in a standardized multi-platform language (ANSI C) and for its ability to create self-contained environments, where each client can run independently.

SAGE-EXPERT uses the whole set of CLIPS capabilities and is, therefore, a complete object oriented Expert System, allowing for the creation of new client applications without requiring new modules for rule processing. Therefore, all development efforts concentrate on acquiring knowledge, writing the rule base and developing an attractive user interface.

SAGE-EXPERT is a native UNIX application, written completely in ANSI C. It can, therefore, run in any UNIX-like operating system (UNIX, UNIXWARE e SUN SOLARIS have already been tested in CEPEL laboratories). Given the fact that its base language is internationally standardized, there is a guaranteed portability, which will not depend on any virtual machine implementation.

Another interesting feature is the fact that any client application, no matter its location in the network, can connect to the server using the well-known socket technology [7] and demand the creation of a stand-alone environment where it can run an Expert System with its knowledge base. The client can, after this creation, send messages to SAGE EXPERT (still using sockets) in order to create new facts, new objects and new rules inside its specific knowledge base, allowing for the execution and chaining of rules, which will be done by the server as in a common Expert System shell (Fig. 1).

\(^1\) Many shells have no application program interfaces (API), which makes it difficult developing practical applications.
The communication is done using a simple protocol based on the XML (extensible Markup Language) standard [8], which consists of a simple set of rules to define text-based messages that can be easily interpreted by computers. The usage of this simple text mechanism, put together with the usage of socket technology for communication, allows for the creation of client applications in any programming language and operating system that the user may find more desirable.

There are many possible applications for Expert Systems to power systems supervision and control. These applications share the incorporation of human knowledge to automatic control center procedures, allowing for time gains, offering faster problem identification and solution, higher performance of human operators, with smaller error rate and knowledge transmission from expert operators to novice technicians.

In the next section we will discuss two client applications that were developed at CEPEL and demonstrate the full capabilities of this framework. These applications are not intended in any possible way to exhaust the whole applicability set of this framework, but yet to show two complete and mature applications of its capabilities to real control center operation.

IV. APPLICATIONS

Using Expert Systems in power systems is not a new idea. Many papers, such as ([9],[10]), prove empirically that Expert Systems are a reliable technique for applications in power systems. Nowadays, their usage has become widespread, but large amounts of time and effort are still being spent on developing the basic capabilities for new applications, an issue that SAGE-EXPERT can address most successfully, as we will now demonstrate.

All capabilities of the SAGE-EXPERT server were empirically validated through the development of two complex applications that can be used in the daily operation of control centers. The applications are:

- SISPRO - an alarm processing system [11]
- RECOMP – an automated system for aiding the restoration of substations after power systems faults [12].

We will proceed to discuss them both in detail.

A. SISPRO - an alarm processing system

In power systems, a single fault can cause multiple alarms and messages. In order to restore the system as fast as possible, and to minimize consumer and the utility company losses, it is fundamental to identify the causes of the faults that triggered the specific alarm set at hand in a fast, safe and precise way [13].

In order to achieve a fast diagnostic, it is expected to use conventional controls and intelligent ones. The intelligent components supplement the conventional controls so as to provide a high level of adaptation to changing conditions and the ability to make decisions quickly by processing imprecise information [14].

Based on the idea of applying the intelligent paradigm to processing the incoming alarms, CEPEL developed an application called SISPRO (Alarm Processing System) that is a client to SAGE-EXPERT and receives all alarms incoming to the SAGE SCADA/EMS system and outputs diagnostics to the root cause of all those alarms. Therefore, it can convert a possibly large alarm set into a single diagnostic that is clearer and allows for faster problem resolution in the power system. Usually, a single fault can cause a large set of alarms and warnings, which, coming together with alarms caused by small routine perturbation in electric equipment, can delay operator response. Having a clear and concise diagnostic minimizes this problem, allowing for quicker operation response and correction actions.

SISPRO is a native application in the SAGE environment and integrates to it seamlessly, without causing any loss in performance, even in the occurrence of an alarm avalanche. SISPRO has been tested in several of those situations and always performed quickly and efficiently, without causing any significative impact on the performance of the rest of the real time system. SISPRO never exceeded 5% of CPU allocation in alarm avalanches, and has a mean allocation of 0.5% when a regular number of alarms happens, with zero CPU usage when no alarm occurred. All data used by SISPRO is real time, current state information, without any delay from the actual arrival at the SCADA system.

SISPRO integrates to SAGE’s alarm visor (Fig. 2), so that the operator can use it and still feel comfortable in the interfaces he is already used to. Since the only change caused by SISPRO is to display its diagnostics on an additional tab in the visor, it is clear that no information is lost and hence the operator can, if he so desires, visualize all the alarms that were caused by the problem at hand. This feature is extremely important in the early days of operation, when the human operator is still growing his confidence on the system. Therefore, having a simple mechanism for on-line human verification will allow the operator to understand the diagnostics and eventually come to trust them, abandoning the long alarm list.
The rules developed for this application were meant to deal with multiple faults simultaneously and, most important, with situations where alarm loss can occur. This loss may be due to communication failures and is not uncommon in SCADA/EMS systems, given the distances between devices and climatic effects. SISPRO deals with alarm losses by assigning a smaller confidence to each diagnostics. When all necessary alarms are received, the confidence assigned to a rule is 100%, but when some alarms are lost, the confidence decreases proportionally.

The probabilities were determined using a data mining system that was validated by the human experts. The data mining system analyzed a data set that consists of a list of independent alarms that have three fields: date/time, equipment and message. The alarms are given in a time ordered fashion, not clustered by any criterion, so we need to start the process by clustering the alarms into faults.

Performing that job manually is not feasible, due to the fact that the history database consists of several megabytes per month. Therefore, we developed a heuristic that can separate the alarms into what we call pseudo-faults. This heuristics separates the alarms based on a concept of equipment neighborhood (two equipments are neighbors if their alarms can occur in a single fault and two alarms can occur in one fault only if they belong to neighbor equipments) and on the time difference between two alarms (two alarms whose time difference is greater than a pre-defined value cannot belong to a same fault, since we are probably talking about two different occurrences in the same equipment).

Once the alarms are separated into faults we apply the data mining algorithm that seeks to extract association rules using the algorithm described in [15] in the format A \rightarrow B in huge data bases, where A \rightarrow B means that if A occurs then we can come to the conclusions that, within a certain degree of certainty, B also occurs. To do that, we take all data sub-sets that occur with a frequency of at least \( f_{\text{min}} \) (where \( f_{\text{min}} \) is a parameter of the algorithm that can be changed according to the user’s needs). Mathematically, this can be written as obtaining the set \( F \) of frequent sub-sets defined as:

\[
F(r, f_{\text{min}}) = \{ X \subseteq R | \text{freq}(X,r) \geq f_{\text{min}} \}
\]

Once separated these subsets are divided in two parts: prior and consequent, such that the value given by

\[
f_{\text{ar}}(X,Y,r) = \frac{\text{freq}(X \cup Y,r)}{\text{freq}(X,r)}
\]

is above a certain minimum threshold. After this step, all frequencies were rounded to the nearest integer and presented to the human experts, who added a few relationships and removed those they considered to be data artifacts, giving us the final set of probabilities to associate with the alarm loss occurrences.

SISPRO is based on templates, that represent patterns, and that allows it to become independent of the actual topology and prevents rule repetition, both for the developer and for the real time system. This allows for a faster and easier development process and improves rule maintainability. Besides, it has a full graphic editor that can be used to tailor the rule base for a specific company and allows for rule reuse in several situations (Fig. 3).

Using patterns simplifies complex projects and allows for knowledge reuse [16]. In alarm processing systems, it increases knowledge representation productivity, because new structures introduced in the power transmission system do not need to be developed from scratch – we can use solutions that have demonstrated in real operation to be representative of the mind models kept by those experts that understand the system. The consequence is a high degree of reusability of the developed solutions that can be used not only for the problem at hand but also as libraries for future solutions.

This characteristic is especially interesting for power systems, because protection and transmission systems are based on a basic set of equipment that follows certain patterns of behavior. Therefore, if two substations have the same configuration, they can share the same pattern, with no need of either rule repetition or new rule testing. Another example: every rule developed for a specific breaker can be reused for all equipment of the same type. This leads to great timesavings and quality increase if we consider that every rule has already been tested for the first breaker. Therefore, no new testing and debugging is required.

Besides the rule editor, a graphical debugging environment was developed. This environment lets the user simulate the occurrence of alarms as if they had come from the real equipment in the real time environment and allow the rule developer to verify if they are correct without having to wait.
for real faults. It has a friendly interface that makes the debugging process easier, and guarantees the correction and reliability of the developed rules. The debugger also has a step-by-step verification feature, allowing the developer to verify if the rule chaining is similar to the one he imagined. This feature also allows for new employees to learn the thought process that leads to the diagnostic. Therefore, the whole system also serves as a library for the knowledge accumulated by the most experienced operators.

Using patterns does not imply in loss of process capability by SISPRO. In extensive tests performed both in CEPEL labs and in real time operation and control centers, all diagnostics were returned faster than the SCADA system can refresh its alarm visor, even in situations where the system was undergoing an alarm avalanche and serving multiple clients.

B. RECOMP - an automated system for aiding the restoration of substations after power systems faults

Restoring a power system after a disturbance is an important problem because power loss can cause economic losses both to the utility company and to its customers. The problem can be defined as determining the best way to take the system from a state where its integrity is damaged due to power faults to another where priority loads are supplied and operational limits are respected [17].

Based on many previous studies and analysis of real contingencies, Brazilian Utilities tend to develop a series of operative instructions (OI) for the operators, where the procedures that must be taken in the case of power losses, either partial or total, are described.

Therefore, whenever a contingency occurs, the operator verifies the OI in order to know what to do, but they are not an ideal solution due to their length and to the huge emotional load operators undergo during system faults [18].

RECOMP (Power Systems Restoration Support System) arose from this need for a computational tool that could help the experts in the critical operation times, showing them the best course of action to take in order to recompose the power system quickly after its disturbance. This tool was developed at CEPEL and decreases the amount of time taken to solve power faults while minimizing the error possibility, assisting the operators in the delicate task of bringing the system back up as quickly and as precisely as possible. Therefore, faster and safer re-energizing brings great benefits to consumers and improves the utility company quality indexes.

Just like SISPRO, RECOMP integrates seamlessly in the SAGE environment. It stays dormant and executes periodically, reading SAGE database and assessing the restoration steps required. When executing, RECOMP usually takes no more than 3% of CPU time, and zero when dormant. Its interface (Fig. 4) uses SAGEs Screen’s Visor with whom the operators are already familiar. This reduces adaptation time and practically eliminates rejection that occurs when new techniques are introduced.

RECOMP is fully object-oriented (OO) and uses all capabilities available in the SAGE-EXPERT environment. Being OO makes it fully reusable in many different transmission and distribution topologies, mostly due to an intelligent inheritance mechanism that allows for a quick and full adaptation to any topology change.

The inheritance mechanism used resulted in a balanced tree, where knowledge is stored in a logical way, making it easier to introduce new equipment types without compromising any of the previous structures. Therefore, new technological developments can be incorporated into the knowledge base, guaranteeing its expansibility and adaptability, without the usual side effects that tend to afflict large rule bases.

Fully graphical editing and debugging environments were also developed for RECOMP. All the knowledge about system restoration must be in the expert system knowledge base and it is necessary to contain the OI. In ordinary expert systems, the developer must update the knowledge base each time the knowledge about the system changes. In Power systems, this can be problematic because the frequent OI changes may cause difficulties in maintaining an up to date system.

In order to improve this situation, the editor is based on object orientation concepts and allows the user to reuse previous classes, so that new developments can be done efficiently without incurring in the same errors that were already eliminated from the previous classes. This is especially useful when the company adds new procedures that are a development from older ones, sharing many characteristics with the previously inserted procedures.

The debugger, shown in Fig. 5 simulates the real time environment and allows for the careful testing of the rules developed. In both cases, the results were similar to those obtained for SISPRO, allowing for easy use and great reliability of the created rules.

The debugger screen emulates, in its bottom half, the real screen seen by users, having the same divisions and layout while reserving the upper half for the insertion of alarms and variable values, as would be seen in the real time environment.
V. CONCLUSIONS AND FURTHER WORK

Both applications were installed in a control center belonging to one of the biggest power companies in Brazil, ELETROSUL, in August, 2005 and showed good performances, aiding the operators in the moments power faults occurred, when response time is a critical variable for the assessment of operation quality.

No application caused significant impacts on the SCADA environment. All other applications in this environment perform with the same efficiency they had prior to RECOMP and SISPRO installation. Neither RECOMP nor SISPRO exceeded 5% of CPU allocation in any given moment, with a mean allocation of 0.5% (zero when no alarm occurs). Besides, their use has no impact on the human-machine interface previously available. Since its integration is done transparently, the resistance that usually affects new computer systems is practically non-existent.

The development of the knowledge base for both softwares represents a high quality electronic documentation that was not previously systematically available. Both in the editors and the run-time environment (where restoration guidelines are shown, if necessary), the user can learn the guidelines, if he wishes.

The framework allowed for thinner clients, which didn’t need to embed a whole Expert System shell. To use this technology, RECOMP and SISPRO only had to know how to communicate through sockets and to process XML messages, both technologies that are highly widespread and usually available through libraries. Therefore, one may conclude that the framework makes it easier to develop new intelligent applications.

Both applications reached the goals of improving work conditions in control centers, decreasing operator response time when faults occurred and serving as corporate memory for training that will be necessary as workforce turnover occurs.

VI. REFERENCES