IMPROVING ELECTRICAL DISTRIBUTION NETWORK OPERATION AND MANAGEMENT AT EDP - CURRENT EXPERIENCE IN SCOPE OF A DA PROJECT

Dora Costa Cabral
EFACEC - Sistemas de Electronica, S.A.
Rua Engº Frederico Ulrich, Apartado 3078, 4470 Moreira da Maia (Portugal)
Tel : + 351 2 9402000 - Fax : + 351 2 9485428 - E-mail : dccabral@se.efacec.pt

António Gomes Varela
Electricidade de Portugel - EDP S.A.
Av. Casal Ribeiro, 50, 5º, 1000 Lisboa (Portugal)
Tel : + 351 1 3531101 - Fax : + 351 1 3537244 - E-mail : gvarela@edinfor.pt

INTRODUCTION

In this document a new architecture is presented, destined to improve the operation and management of electrical distribution networks. Thus the main feature of this architecture is a loosely coupled event bus allowing the various applications used in a distribution centre control room to be effectively integrated.
The document then moves one of those principal areas of functionality to be expected in a modern control room, the management of safety documents and the implementation of safety rules during the day to day running of the network.

OVERVIEW

EDP Group General Overview

EDP - Electricidade de Portugal, S.A. is, presently, an entrepreneurial group composed of different companies. It's activity is carried out in all the mainland and covers the generation, buying, transmission, distribution and selling of electrical energy in Portugal.
EDP is now divided into several utilities, one related to the electrical Power Generation, another dealing with High Voltage network Transmission (above 150 kV) and the remaining ones related to the operation of the Distribution network, with 5 million customers.

EFACEC General Overview

EFACEC Sistemas de Electrónica, emerged as an independent company in 1991. Its prime markets are those of Information Technologies and Electronics. Its part of the EFACEC Group, which was formed in 1940 and is at the moment the largest Portuguese Group in Electricity and Electronics. EFACEC Sistemas de Electrónica with the head office in the north of Portugal, covers the following main business areas:

- Energy Systems Automation and Telecontrol
- Transport Systems
- Power Supply Systems
- Installation and Services
- Electronics Production

Business Driven Overview

In general terms, electrical utilities are facing a challenge to reduce life-cycle cost, increase customer value and improve flexibility in operation and services. Both Distribution Automation (DA) and Demand Side Management (DSM), when implemented will help to meet the above mentioned key aspects. This will lead to power losses reduction, increasing reliability, reducing the need of maintenance and giving the Customer the opportunity to select between different tariff schemes.
The development of a business case that recognises, estimates and compares the value of benefits against anticipated implementation costs is a pre-requisite for any utility decision in order to proceed with a major Automation project.
EDP´s trend in Automation business case development was the identification of the systems that will provide quick returns on investment and/or improve customer service and satisfaction.
Automation, at a top level, means integrating networks, databases and client applications in all EDP´s technical systems. An exhaustive study was made, on the different network levels in order to identify the highest benefit location and integration to maximise the benefits and minimise the costs of implementation. These benefits were recognised by EDP Distribution utilities.

SYSTEM ARCHITECTURE OVERVIEW

The Collaborating Component Architecture (CCA), envisions software entities called "components" collaborating in a distributed computing environment. The components may be seen as functional units, implementing a full or partial application, and usually they co-operate with each other to accomplish a full functionality to the end-user. To provide this co-operation there is the need to support an adequate communication model with a clear contract / interface independently of the machinery or network environment within which the components run.

For the type of system that primarily function in a event-driven and exception oriented environments an appropriate communication paradigm is that of de-coupled messaging that allows for functional independence, loose coupling of components and asynchronous communications.
This enables a component that is an originator of data to share it with other components without being explicitly aware of who are the potencial recipients of the information. The same way the consumers do not know from where the information comes and can subscribe for the data they are interested in without affecting the components that publish the data. This briefly described Publish / Subscribe model is the foundation for collaborative functionality. The components are linked together by a software bus and the contract they have with the rest of the world is supported by an Event Model.

Technology issues and solutions

The architecture of a modern system must be aware of the new emerging technologies. In the recent years OOT - Object Oriented Technology, is playing an increasingly major role in software development. One of the most important factors to go for a new technology is the definition of standards and it is noticeable that there has been a significant evolution in this field, with the creation of international organisations and consortiums with the objective of defining and promoting OO standards. Being so it was decided to follow an OO approach as the basis for supporting the CCA foundations.
Although there are other alternatives, the architectural spearhead of OO at the distributed system level and interoperability is the Object Management Group (OMG) CORBA standard. Convergence to CORBA is a trend, and the best strategy is to build the CCA consistent with CORBA so that it can take full advantage of CORBA-compliant proven commercial products.
The software bus links all the components together, without those components being aware of the hardware, network environment and any internal bus layers. Thus the component view is simply as shown on Figure 1.

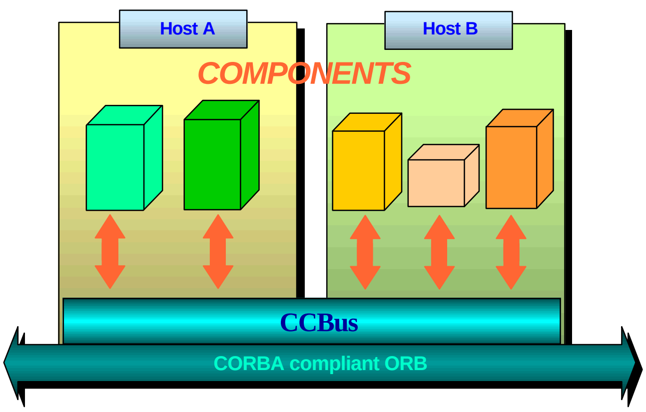


Figure 1 - Collaborating Component

CCA however is targeted to the utility problem domain. As such it may be necessary to create some specialisation on top of the general view of components taken by CORBA. That is the role of the CCBus, to provide a layer of code that isolates components from the direct usage of CORBA and provides a common and consistent framework to the development and runtime environment of a component.

The Distribution Event Bus

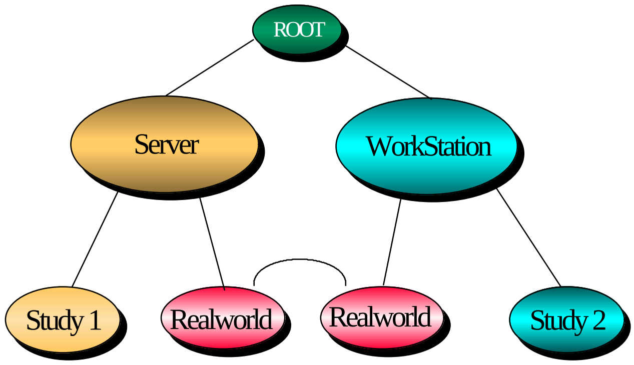
This section further expands the main software concepts of the distributed event bus. The main topics covered are the components, contexts and the tools / components closely related with the bus.

Components

A component is the fundamental unit of both functionality and distributability. A component may be built so as to function as data repository, control algorithm, report generation, gateway to legacy system, power system model part, service, or whatever else is required. Any running component is actually an “instance” of a component type, and there may be many instances of the same component type active, either in the same host or elsewhere in the network.
A typical life cycle for a component, after being initialised, is to collect events from the bus, process them updating its internal models, produce any public information to other components publishing events and return to a “waiting for events” state. Other components process by entering a state where they can interact with agents outside the world of collaborating components (i.e. outside CCA).

Contexts

Topics and contexts are the basic mechanisms providing information sharing and joining components over related functionality. However, as much as they provide sharing they should also provide separation, keeping components apart - different instances of the same component type will be required to run concurrently but perform for different jobs, such as distinguishing real world and study environments.
The topic is the elementary way components share information. A component may publish and / or subscribe data on more than one topic. A context can be seen as an aggregation of topics built up from the components running under the same context. Usually a component instance runs in a single context but it may have links to other contexts.



Contexts are organised in a hierarchical structure. The events published in the upper contexts are routed down to all components registered for subscription in the lower contexts. Thus data services operate at the top of the hierarchy and active models towards the bottom. A data request to load a RealWorld component, for example, could rise to a data service in the Server or ROOT. Any data changes published from the data service in the ROOT would arrive at any subscribing component in any of the contexts illustrated.

The Importance of Contexts

The components presented in the following sections may be used in the RealWorld, to assist the operator executing already planned work, and in emergency situations where no previous planning has been possible. The same components may be used in studies, to plan future work or to explore alternative network configurations and feeding patterns. Without the support of this type of bus and the concept of contexts the component code would become heavy with distinctions being made between study and RealWorld usage, between online and hot-standby functions. Such code has been known to overpower the functional code within system. With the support of contexts and the loose coupling provided by the bus the components run without being specifically modified for each situation.

The Components

The following sections present three areas of functionality well served by the concepts surrounding the bus, all of these areas being connected with safety.
The first of these sections talks briefly on the improved presentation of information to the user, and though this may have more to do with the RealWorld, because of the loose coupling provided by the bus, can easily be used in a study where small components publish the flow information required, either by collecting power application results or simply following manually applied actions.
The second section covers the safety rules, or any other company rules applicable by software, covering the day to day running or exploitation of the distribution network. The final section covers the document management used to manage the switching and field crews, centred around the switching order or switching schedule, i.e. based in the existing methods but linking them with the dynamic network model. This model is presented to all operators and other system users, clearly presenting the activity in progress on the network with lists, diagrams and operator messages. This model thus not only allows the actions of each dispatcher / field crew combination to be checked against the safety rules but also provides the means of communication directly between these combinations, warning when the actions of one are likely to influence another. Again, without the flexible concepts of the bus to support these tools, there implementation becomes more complex, there added value reduced.

VISUALISING THE NETWORK

Visualising the network is a very important issue, the most frequent activity and the best opportunity of providing information to the control room users. If a lot of information is displayed simultaneously then the user will lose the knowledge, the clear picture of what is going on and this can lead to a mistakes.

Network Colouring

The information displayed at any moment is configurable. The user can define a set of visualisation styles with the information that he requires and change from one style to other without any problem. He can create styles to visualise voltage levels, galvanic connections, energisation, earthing. In any of the styles used the operator will always have to its disposal the on-demand tracing to highlight feeding configurations or similar dynamically changing queries.

Colouring by injection point

Colouring by injection point allows the user to continuously visualise, using colour, the flow of power from one or more parts of the network, normally substations, to other parts of the network, normally other substations, switchyards or distribution transformers. Ideally any part of the network can be identified for providing a source for the flow and / or showing the receipt of the flow.
The colour flow follows the flow of power and is visualised only where the user (operator or map designer) specifies this.
The direction of the flow is determined according to the situations, being able to use SCADA values, state estimated results or others.

SAFETY RULES

Safety is one of the most important issues for any company and the distributors are not an exception. Safety with persons and with the network itself raises several questions, only some of which can be addressed by a computer system. Thus the following functions accept that the user remains the final arbiter on safety issues but provide powerful help in this area.
A set of safety rules and procedures is proposed a to help the control engineers and field engineers manage the network.
The operator will have the ability to better understand the network and to check the network reaction to his actions, normally before the actions are "committed". He will always have an "assistant" to alert him to any detected risk that can occur but the operator always gives the final decision

The General Principles

The rules implemented are based on the exploitation and procedures manuals of EDP but they are very configurable in order to easily adapt to any other distribution company.
The assistance covered falls broadly into the following areas :

- Providing improved diagrams and reports to both the control room and field crews
- Providing the ability to track the presence of field crews on the power network and their activity
- Providing the ability to place standard and freeform tag and document labels onto the electronic diagram and to bring such information to the users attention at appropriate times
- To never allow the system to refuse a users instruction, but to ensure that any detected risk is brought to the users attention and logged if ignored or overridden
- To allow the user to prepare and verify switching activity and network configuration changes in advance in a study environment or context
- To check all instructions and network changes known to the system and warn where the actions or reports generate a risk, whether for the crews directly concerned or for other crews / people known to the system
- To allow additional checks to be manually made where online information is not directly available, e.g. check the capability of a switch to break an unmeasured, unestimated flow
- To allow the system to be configured in such a way that the user interface has a consistent look and feel in all modes / applications but clearly deviates from this normality when presenting abnormal or threatening situations to the user

The safety rule model is permanently available to ensure that the rules and procedures as defined are checked *at* all times and to ensure the appropriate users are alerted to any violation or potential violation of these rules.

Rules / Procedures

The system has two different levels of rules/procedures. Within this presentation we refer to basic rules that are known and understood by the model and user defined rules that are defined by the user and associated with some action which the model does not directly recognise.

Basic Rules. The basic rules are simple conditions, most of them are related, but not limited to, the definition of the safe regions for field crews. They are intended to assist in operations like:

- Isolation of an equipment from all the possible injection points
- Lock the surrounding switches in an open state
- Check that the part of the network where the work will occur is dead
- Assure that all the necessary earth connections are made

These conditions are capture in the model and are the base for all safety.

Dynamic Rules. The dynamic rules are conditions defined by the user. They can be a set of basic conditions or a condition that the model can validate because it is told how to do it but does not really understand what it is doing.
The user will be able to define an action to be made that contains pre or post conditions. In instructions like this the validation of the action is dependent on the validation of all the conditions, in the order defined, otherwise the action will be reported as an invalid one.
These rules are very useful in the specification of works orders or switching orders.

Active Model

The active model will be responsible for ensuring that the rules defined aren't broken.
It will validate each operation on the network and alert the user when something he is doing goes against those rules.
The rules are configurable and will include rules ranging from checking that the value of an attribute of an equipment is inside a defined range, or check if an attribute is present in the equipment (e.g. check if the attribute live is present in the equipment) to ensuring that conflicting attributes do not exist in the same island, e.g. a test certificate and a work permit.
It is of interest to maintain the individual rules as simple as possible, to be sure that the user will always understand what he is doing, what the system is reporting and to ensure that the system works properly. There is no restriction to the number on conditions associated to each

instruction and the result will depend on the result of all the conditions.
Each condition will have a set of states associated with it and the user will also be able to define which state or states for a condition are considered valid.
As already been stated the model will not inhibit any action of the user, it will only alert the user of defined conditions that have failed, indicating the condition that is failing. It will be the users responsibility to decide to continue or not. The system shall log the overrides of any warning.
If in any situations the system is incapable of checking if an instruction is safe or not, the active model will alert the user to the situation, leaving to the user the final decision as always.

SWITCHING ORDERS / SCHEDULES

When we think in operating the network the switching orders become the important focus of all work. The system provides facility to the user to propose and execute, as a switching order, a sequence of operations on the network, either with the aid of telecontrol or on-site field staff.
The switching order can be created and planned in advance of its proposed execution. Alternatively it can be created and executed immediately on-demand, e.g. as required for a fault job.
With a correct definition of the safety rules the system will be very useful in the creation and validation of those sequences, whether on demand or planned.
A switching order, like any other document, passes through a set of user defined states. States including in-preparation, prepared, authorised, executing, abandoned or completed are some of the states possible. The system activities permitted may be checked against the document state.

Preparing a switching order

The preparation of a switching order includes the definition of a set of instructions to be performed , associating them to the necessary validation conditions that will need to be checked. The order of those instructions also needs to be verified once the validation of an instruction depends on the execution and validation of the previous ones.
One useful functionality is the ability of associate a set of instructions, those instructions will work as a block and its validations will be made at a block level. The validation of a block will reflect the validation of all its instructions. This supports the normal situation where a field crew is given a sequenced block of instructions to execute before reporting back to the control centre.
The capability to automatically generate switching instruction sequences based on a configurable set of aggregated instruction rules (e.g. isolate and earth a feeder section) is also to be provided. Thus routine jobs take less time and less resources. Complementing this activity is the facility to ‘invert’ a sequence of instructions, allowing a reverse order, reverse meaning sequence of instructions to be created. Obviously this is targeted to providing restoration sequences based on the original isolation and earthing procedures. Each instruction or block of instructions in a switching order will have associated with it a state. The cycle includes the basic proposed /validated, instructed and confirmed. Additional states also exist, e.g., timed out, cancelled for conditions outside of the normal sequences. The state of the switching order and its instructions is visible to all users, allowing any user to know what is happening at any time.

Safety documents

Safety documents are used to manage the authorised safe access to and type of work that may be carried out by field staff.
The types of documents available in the system are configurable, being able to have for example Permits to Work, Sanctions for Test, Limitation of Access, Record of Inter System Safety Precautions, Safety Declaration, etc.
Those documents can be associated with equipments or parts of the network, with that information being displayed on the diagram. The user can have available on the diagram all the works that are planned or being executed in the network and the state of those works also. Such a focused symbology is a good example of how multiple symbologies allow information to be displayed without overloading the dynamic colouring or other symbolic aspects.
It is possible to have inside a document a reference to other pre-defined documents. Any actions on the one document may check the state of the related documents, permitting or otherwise the document progression.

Executing a switching order

During the execution of a switching order the system will report to the user feedback on the user actions and the system checks. These checks will be executed whether the switching was planned or is on demand.
In cases where safety checks are reported as failed to the user or where a safety conclusion could not be reached, the system will alert that fact and the user will be left with the final decisions. In any cases it is always the user that decides what to do, even if the system reports an instruction as an invalid one if he user decides to, the instruction will be executed. The system will only log the override.

Multiple field crews working

A good example of where the system may be unable to accurately check the safety of blocks of instructions is when multiple field crews are working simultaneously in adjacent parts of the network. Situations can occur where the system can not evaluate if a condition will be valid or not because the pending or current activity of one team leaves the network concerned in an unknown state until the switching is confirmed. That is to say, between the issuing of the instructions to the field crew and their confirmation that the switching is complete, the system is unaware of the local network state and can not always accurately check adjacent activity. Of course, the system could exhaustively check all the possible combinations of the actions of the two teams but that could be very onerous. Therefore the system will only report that it is not able to verify the safety of the parallel activities, only the safety of each activity on its own.

CONCLUSIONS

The integration of all the various software packages within an electrical distribution control room, within the company, is an important part of the process that these companies are undergoing. Getting the right balance between monolithic systems and separate, unintegrated departmental systems is a challenge that must be faced in this fast changing business.

The bus described here is intended to allow the companies to manage their own integration.

The use of such a bus to link the control room applications described, whether these applications are running in active RealWorld or in a passive study, provides an open and flexible system. Where a collection of applications can provide a unified user interface.

Amongst the applications that can be deployed within this environment must be included those applications that manage the day to day running of the network. Foremost amongst these applications must be the visualisation and organisation of the network state and its associated switching and engineering works. These features enhance the safety of the network management by raising the user interface to the level of an ever present assistant with consistent behaviour throughout all online and planning activities.

REFERENCES

[1] Gomes Varela, Alberto Rodrigues, “Integrated Real-time and Non Real-time Applications for Information and Business Exchanges - A joint EDP / EFACEC Project”, in Proceedings 1998 DA/DSM

[2] J.P. Britton, "An Open, Object-Based Model as the Basis of an Architecture for Distribution Control Centers", IEEE Transactions on Power Systems, Volume 7, Number 4, November 1992, pp. 1500-1508.

[3] David Marsh, “The Definition Provision and Support of an Event Service”, EDP / EFACEC response to OMG Utility Special Interest Group RFI - March 98