Control Architecture for Substation Automation Systems based on IEC 61850 and IEC 61499 Standards

VALENTIN VLAD, CEZAR DUMITRU POPA, CORNELIU OCTAVIAN TURCU, CORNELIU BUZDUGA

Electrical Engineering and Computer Science Department
Ștefan cel Mare University of Suceava
Str. Universității, 13
ROMANIA
{vladv | cezarp | cturcu | cbuzduga}@eed.usv.ro

Abstract: This paper proposes a control architecture for the substation automation systems, combining the artifacts of the IEC 61499 standard, for modeling and implementation of the control logic, with the specifications of the IEC 61850 standard, for communication and information exchange between the automation devices. The proposed control architecture is implemented and validated in a simple fault protection scenario.

Key-Words: substation, automation systems, IEC 61850, function blocks, protection, power systems

1 Introduction

Recent advances in substation automation systems include the adoption of a new standard for communication between intelligent electronic devices (IEDs), published by the International Electrotechnical Commission (IEC) under the name IEC 61850. The standard specifies how the data should be organized in IEDs, defines a list of services for reading and modifying these data, and proposes a mapping of the data and services to a specific communication protocol [1], [2]. However, the standard do not impose a specific type of implementation for the control logic within IEDs, nor how different control components in the same IED should communicate with each other, but rather the external behavior of the automation devices. Different implementations for the communication interface defined by the standard exist, commercial or open source, in different programming languages, like C and Java. There are also proprietary implementations, included in commercial protection and control devices for substation automations [3], [4].

A convenient solution for developing distributed and reconfigurable control applications is the one proposed by the IEC 61499 standard, which is focused on applying function blocks in the design of distributed industrial-process measurement and control systems. IEC 61499 includes advanced software technologies, such as the encapsulation of functionality, component-based design, event-driven execution and distribution, and is considered a complementary standard to the IEC 61131 standard for PLCs.

In this paper we propose a control architecture for substation automation systems, which combines the IEC 61499 models, for the implementation of the control logic, and the IEC 61850 specifications, for communications. The reasons for choosing the IEC 61499 standard for modeling the control logic are mainly related to its comprehensive methodology of building control applications from modular components, in a graphical way. The scientific literature contains also published work on harmonizing the IEC 61850 and IEC 61499 standards, the representative papers including [5], [6] and [7].

The rest of the document is organized as follows. The next chapter gives an overview of the IEC 61850 and IEC 61499 standards. The third chapter introduces the proposed control architecture, and the fourth one presents its implementation and validation in a simple fault protection scenario.

2 IEC 61850 and IEC 61499

2.1 IEC 61850

IEC 61850 is a relatively new standard (2004) designed to improve the communication between the automation devices within substations. In contrast with the legacy protocols, focused on how bytes are transmitted on the wire, the standard deals also with the internal organization of data in
devices, to achieve a better interoperability and reduced configuration costs [1].

The standard adopts a model-driven approach by standardizing device, object and service models. A physical device (an IED) is typically defined by its network address and contains one or more logical devices. Each logical device contains one or more logical nodes (LN), which are named grouping of data and associated services, logically related to some power system function (Fig. 1). Different types of logical nodes are defined, e.g. for automatic control, for metering and measurement, for supervisory control, protection, switchgears, etc. The name of a logical node instance consists of a standard part of four letters (e.g. XCBR, for circuit breaker control) and a suffix, representing the instance ID (e.g. XCBR1, XCBR2). An optional, application specific prefix can also be used as part of the LN instance name. Each logical node contains one or more elements of Data, with names and types defined by the standard (e.g. Pos, in case of XCBR LN types). Each element of Data within the logical node conforms to the specification of a common data class (CDC), describing the type and structure of the data. The data attributes of the common data classes contains in fact the logical node data (e.g. the data attribute stVal of the Pos element of Data in an XCBR LN indicates the position of the circuit breaker).

The abstract model of the device and the communication services are mapped to a specific communication protocol stack based on MMS (Manufacturing Messaging Specification – ISO 9506), TCP/IP, and Ethernet.

As to the substation architecture, the standard places the monitor and control equipment on three levels, namely process, bay and substation level, as illustrated in Fig. 2. The devices at the process level are designed to collect information such as voltage, current, and status information from the primary equipment and to transmit them in a digital form to the upper level. The bay level includes IEDs running applications for protection and control, while the substation level is dedicated to applications for monitoring and control of the whole substation.

![Substation architecture according to the IEC 61850 standard](image)

**Fig.2. Substation architecture according to the IEC 61850 standard**

### 2.2 IEC 61499

IEC 61499 defines an open architecture for distributed control and automation, presented in terms of implementable reference models, textual syntax and graphical representations. The programming unit of the IEC 61499 is the function block (FB), from which complex control applications can be built. The standard defines three types of function blocks: basic, composite, and service interface function blocks. Basic function blocks perform elemental control functions, such as reading a sensor or setting the state of an actuator by executing various encapsulated algorithms, according to an execution control chart. The functionality of composite function blocks is determined by a network of interconnected function blocks inside. The service interface function blocks (SIFB) serve to abstract the specific functions of a hardware platform, allowing the application developer to focus on the application logic.
The control applications are built (according to IEC 61499) as networks of function blocks, which can be distributed across several devices. The IEC 61499 device model contains one or more resources, to which parts of a function block network can be mapped, to be executed. According to [9], a resource is considered “a functional unit contained in a device, which has independent control of its operation. It may be created, parameterized, started up, deleted, etc., without affecting other resources within a device.” This definition encouraged us to assimilate an IEC 61850 logical node to IEC 61499 resource to, as explained in the next section.

3 The proposed control architecture

As mentioned in introduction, our approach for implementing control applications for substation automation relies on IEC 61499 models for the control logic and on IEC 61850 specifications for communication and information exchange between IEDs. As illustrated in Fig. 3, we propose a logical node be modeled and implemented as a network of function blocks mapped to an IEC 61499 resource. The function block network will contain a principal function block encapsulating the data of the logical node (as specified in the IEC 61850 standard) and function blocks for internal communications (with logical nodes in the same device) or for interfacing with the primary equipment. The principal function block of a logical node will receive information from the process or from other logical nodes and will generate actions (e.g. sending messages) along with updating its internal data, according to the encapsulated control algorithms.

As to the communication part, an IEC 61850 server should be implemented as specified in part 7-2 of the standard. The server will exhibit outside the data indicated in the configuration file (in the format defined by the Substation Configuration description Language – SCL), updated continuously from the function blocks implementing the control logic. The synchronization of data between the control logic and the server can be realized through a modified IEC 61499 remote device, including functions for accessing data in principal function blocks, based on their IEC 61850-compliant references. This mechanism, along with the control architecture, is illustrated in Fig. 4, considering as example a breaker IED.

![Fig.3. Logical nodes modeled as networks of function blocks, mapped to resources](image)

![Fig.4. Proposed control and communication architecture (example for a Breaker IED)](image)
The transmission of GOOSE (Generic Object Oriented Substation Events) messages or sampled values using multicast can be realized with specialized service interface function blocks, transmitting the data at regular intervals or when requested by principal function blocks.

4 Implementation aspects
The proposed control architecture was implemented and tested in a simple fault protection scenario, with simulated power equipment. The protection scenario (illustrated in Fig. 5) involves the transmission of GOOSE messages between several IEC 61850 logical nodes for tripping a circuit breaker and for publishing its new position.

The values measured by the current transformer (CT) are transmitted to a PTOC (protection overcurrent) logical node. When an overcurrent is detected, PTOC communicates the anomalous condition to the PTRC (protection trip conditioning) LN, which issues a trip command (in form of a GOOSE message) to the XCBR LN. As a result the circuit breaker (CB) is open and the new status is transmitted (also through GOOSE messages) to the PTRC and RREC (auto-reclosing) logical nodes. After a short time, RREC issues a reclose command to the XCBR LN, which closes the circuit breaker and publishes its new status. The current through the feeder can be modified through a user interface simulating the power system, presented in Fig. 6.

The control logic was implemented as an IEC 61499 system with five devices, as illustrated in Fig. 7. The function blocks modeling the logical nodes were encapsulated in IEC 61499 resources, named according to the syntax defined by the IEC 61850 standard for the logical node names. For each device, an IEC 61850 server was started, assuring a communication interface described in a SCL file. It can be noted that there must exists correspondence between the data type templates defined in the SCL file and the data implemented in the FB-based logical nodes. For the implementation of the IEC 61850 server we used the open source library presented at [4].

Acknowledgment
This paper has been financially supported within the project entitled „SOCERT. Knowledge society, dynamism through research”, contract number POSDRU/159/1.5/S/132406. This project is co-financed by European Social Fund through Sectoral Operational Programme for Human Resources Development 2007-2013. Investing in people!”

References:


