TU WIEN Informatics

Framework for the Design and Automatic Deployment of Smart Grid Applications

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Technische Informatik

eingereicht von

Felix Knorr, BSc
Matrikelnummer 01325541

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dr. Wolfgang Kastner
Mitwirkung: Dipl.-Ing. Thomas Frühwirth

Wien, 1. Mai 2021

Felix Knorr	Wolfgang Kastner

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

TU WIEN Informatics

Framework for the Design and Automatic Deployment of Smart Grid Applications

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Computer Engineering

by

Felix Knorr, BSc
Registration Number 01325541

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Dr. Wolfgang Kastner
Assistance: Dipl.-Ing. Thomas Frühwirth

Vienna, 1st May, 2021

Felix Knorr	Wolfgang Kastner

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Felix Knorr, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. Mai 2021

Felix Knorr

Danksagung

Mein Dank gilt zuallererst Prof. Wolfgang Kastner, der mich nicht nur mit seinen Vorlesungen auf diesen spannenden Teilbereich der Informatik aufmerksam gemacht hat, sondern mich auch in seinem Institut willkommen geheißen hat.

Weiters möchte ich mich bei meinem Betreuer Thomas Frühwirth bedanken, der mir immer mit Rat und Tat zur Seite gestanden ist, auch wenn es noch so stressig war. Auch möchte ich mich beim gesamten ASG für die angenehme Arbeitsatmosphäre, sowie die netten diskussionsreichen Mittagspausen bedanken, die eine willkommene Abwechslung dargestellt haben.

Auch bei meinem Arbeitgeber Siemens möchte ich mich bedanken, der es mir ermöglicht hat, Studium und Arbeit unter einen Hut zu bringen. Hier speziell bei Lukas Krammer, der mir geholfen hat, den Kontakt zu Prof. Kastner herzustellen und Alfred Einfalt, der mich benefalls stets mit fachlichem Rat unterstützt hat und mir die Freiräume für die Diplomarbeit gewährt hat.

Letztendlich möchte ich mich noch bei meinen Eltern für die finanzielle und emotionale Unterstützung während des gesamten Studiums bedanken.

Acknowledgements

First of all, I would like to thank Prof. Wolfgang Kastner, who not only brought this exciting area of computer science to me, with his lectures but also welcomed me to his institute.

Furthermore, I would like to thank my supervisor Thomas Frühwirth for many insightful conversations during the development of the ideas in the thesis, and helpful comments on the text, even when he was busy himself. I would also like to thank the entire ASG for the pleasant atmosphere and the nice discussions during lunch breaks.

Likewise, I would like to thank my employer, Siemens, who offered me a perfect study work balance. In particular, I would like to thank Lukas Krammer, who helped me to establish contact with Prof. Kastner, and Alfred Einfalt, who supported me with professional advice.

Finally, I would like to thank my parents Christine and Günther for their financial and emotional support throughout my studies.

Kurzfassung

Der Energiesektor erlebt einen umfassenden technologischen Wandel. Durch die immer größer werdende Verbreitung intelligenter Geräte sowie verteilter Energieerzeuger wird aus dem vormals hierarchisch strukturierten Stromnetz ein dynamischer Verbund aus Stromerzeugern, Speichern, elektrischen Verbrauchern und Netzbetriebsmitteln. Dies bringt nicht nur Vorteile sondern auch große Herausforderungen mit sich, die besonders bei der Entwicklung neuer Anwendungen zum Tragen kommen. Gängige Entwicklungsmethoden in diesem Bereich sind nicht für Applikationen dieser Größe und Komplexität gedacht. So existieren zwar bereits durchgängige Tools für die Entwicklung neuer Smart Grid Anwendungen, diese erlauben jedoch nur die Beschreibung einer Anwendung für ein spezifisches Smart Grid. Das Ziel dieser Diplomarbeit war die Entwicklung eines durchgängigen Frameworks, welches eine vom Smart Grid unabhängige Beschreibung von Applikationen erlaubt. Dadurch wird ein Mehraufwand vermieden und einheitliche Lösungen werden gefördert.

Das entwickelte Framwork basiert auf den offenen Standards IEC 61499 und IEC 61850. Während IEC 61850 eine einheitliche Systembeschreibung von Smart Grid Komponenten erlaubt, bietet IEC 61499 ein Modell zur Entwicklung verteilter Anwendungen. Die Idee des entwickelten Frameworks ist, IEC 61850 Funktionalität in Funktionsblöcken aus IEC 61499 zu kapseln, um anschließend neue Applikationen mit Hilfe dieser Funktionsblöcke beschreiben zu können. Das Framework besteht aus drei Modulen. Neben einem Modul für die Beschreibung der Anwendung mittels Funktionsblöcken ist ein weiteres Modul für die Verteilung dieser Funktionsblöcke auf vorhandene Geräte verantwortlich. Dabei ist es möglich, Bedingungen zu definieren, um abhängig davon beste Lösungen zur Verteilung zu finden. Desweiteren wird erst in diesem Schritt der Aufbau des jeweiligen Smart Grids bekanntgegeben. Im letzten Schritt werden dann, abhängig von der gewählten Lösung, automatisch die jeweiligen Konfigurationsdateien der einzelnen Geräte generiert und die entworfene Applikation in die dafür ausgewählten Geräte geladen.

Anhand eines spezifischen Anwendungsfalls wurde im Anschluss die Funktionsweise des Frameworks evaluiert. Zusätzlich wurde es in eine virtuelle Testumgebung eingespielt. Die Arbeit schließt mit einem Ausblick auf nächste Schritte, die für den produktiven Einsatz des Frameworks noch umgesetzt werden müssen.

Abstract

The energy sector is undergoing a fundamental technological transformation. The increasing prevalence of intelligent devices and distributed energy resources is changing the previously hierarchical structure of the power grid into a dynamic interconnection of power generators, storage units, electrical consumers and network resources, altogether known under the umbrella term smart grid. This development not only leads to many advantages, but also to major challenges. Especially the implementation complexity of new applications rises significantly. Traditional development methods in this area are not designed for applications of this size and complexity. For example, although there are already consistent tools for the development of new smart grid applications, these are limited to the description of an application for a specific smart grid. The idea of this thesis was the development of an integrated framework, which allows a description of applications independently of a given smart grid. This eliminates unnecessary work and promotes uniform solutions.

The developed framework is based on the open standards IEC 61499 and IEC 61850. While IEC 61850 allows the uniform system description of smart grid components, IEC 61499 provides a model for developing distributed applications. The idea of this framework is to encapsulate IEC 61850 functionality in function blocks from IEC 61499 to describe new applications and use them. The developed framework consists of three modules. Besides a module for describing the application using function blocks, a further module is responsible for distributing them to an existing device infrastructure. Thereby, it is possible to define constraints in order to find best solutions. Moreover, the structure of the smart grid is only specified in this module. Depending on the selected solution, the respective configuration files of the individual devices are generated automatically, which are then used for deploying the application to the field.

Using a specific use-case, the functionality of the framework was evaluated. In addition, it was deployed to a virtual test environment. Consequently, the feasibility was demonstrated. The thesis concludes with an outlook on next steps that still need to be implemented for the productive use of the framework.

Contents

CHAPTER 1

Introduction

In the past, energy infrastructure was characterized by a strictly hierarchical, top-down architecture, where large-scale power plants, such as hydro and fossil fuel power plants, were responsible for the production of energy, which was then transported through different voltage levels (extra high, high and low voltage) to the end customer [tec11][ABS]. The integration of Renewable Energy Sources (RESs) and Distributed Energy Resources (DERs) such as wind turbines and Photovoltaic (PV) systems, but also the on-going massive roll-out of smart meters and other (smart) devices capable of interacting, has not only opened many new possibilities but also many challenges [ABS][CKKL17]. For example, by installing PV systems, end customers can become energy producers as well. Therefore, they are often called prosumers in literature [ASK14][GT]. Besides efficient and sustainable energy systems, the desired properties of the future Smart Grid (SG) are self-healing capabilities, adaptiveness, and the support of islanding [SKG16]. But of course, the coordination of the bidirectional flow of energy and information as well as the stochastic nature of RESs (energy fluctuation) require sophisticated new concepts, which are best supported by Information and Communication (ICT) technologies [ASS].

Due to the increased complexity of application development in this domain, a new framework is proposed in this thesis. This framework supports the different actors in the planning, development and deployment of new applications. As a proof of concept, the framework will then be used to model parts of the Fault Location Isolation and Service Restoration (FLISR) scenario. The aim of this important SG use-case is the automatic reconfiguration of the grid such that faulty feeders can be bypassed. It is also an important concept for supporting islanding.

1.1 Smart Grid

According to the European Union Commission Task Force for Smart Grids, a SG is defined as "an electricity network that can cost efficiently integrate the behavior and

actions of all users connected to it – generators, consumers and those that do both – in order to ensure economically efficient, sustainable power system with low losses and high levels of quality and security of supply and safety" [BS20]. But even if a specific definition is given here, there are many different interpretations of what a SG is. For example, Stuart Borlase states that the term SG rather describes the "essential change the way we address the energy demand" [Bor16]. There are several causes for this change. For example, the change to RESs in the context of the 'Energiewende'. But also the reduction of Europe's dependence on oil- and other fossil-fuel-producing countries plays an important role, as well as decreasing availability of fossil-fuels in general and the resulting price increase. [BS20][Bor16]

In addition to benefits for network operators such as increased reliability through improved error detection and self-healing, increased efficiency, and improved sustainability, there are also direct benefits for the end customer. A SG could result in reduced bills and better control, for example through flexible pricing, but also real-time information on consumption and costs. Of course, this also requires a greater flow of information from the end consumers to the network operators. Data protection is therefore an extremely important issue. Additional information about the data protection problems caused by the introduction of smart meters in Austria, as passed by the EU can be found in the following papers by Cejka et al. [CKKL17][CKK19]. [Bor16][Mom12][Dil20]

The transition to SGs is also leading to a paradigm shift on all levels of the electrical infrastructure and the market. Whereas in the past mainly centralized concepts were pursued, nowadays the trend is moving towards more decentralization. For example, instead of a few large scale power plants, many small renewable energy power plants are responsible for the production of energy. These can be local wind parks or even PV systems on the roofs of consumers. As a consequence, also large power lines and pipelines can be downsized. However, decentralization also causes a higher complexity. The former passive consumers are now active participants who can generate and supply power to the network. [Atl18]

The Power System Cognification (PoSyCo) project funded by the Austrian Research Promotion Agency (FFG) tries to solve some of the mentioned challenges, whereby it focuses on the so-called SOFTprotection of future SGs. SOFTprotection, additionally to existing safety features, is a system, which will enable many different use-cases like intelligent overload prevention and reporting and recovery functionality in cases of faults. This new system is needed because conservative approaches will not work as expected when considering bidirectional load flows. As previously described, in a SG, customers can act as prosumers. Therefore, it would not be sufficient for a fuse located at a (secondary) substation to break a faulty feeder if "generators along the feeder strongly contribute to the fault current" [oT15]. Expected results of PoSyCo are not only the implementation of such a SOFTprotection system but also the design and implementation of advanced SG applications. [GT][oT15]

1.2 Problem Statement

Traditional engineering methods were not intended to be used for applications of this size and complexity. Therefore, the International Electrotechnical Commission (IEC) and other standardization organizations have worked out many new standards[1] for the SG domain. However, there are still many gaps in between. In particular, there is no common tool or framework that supports the system engineer from design to deployment. Existing approaches follow the schema of first selecting the infrastructure and components and then trying to implement the desired functionality, i.e., the use-case, on it. However, there is currently no way to just functionally describe the use-case (and the already existing infrastructure) to then automatically deploy a solution to the devices in the field. As a consequence, applications need to be remodeled every time to apply them to different SGs. [PA]

The goal of this thesis is to develop a generic framework for the design and implementation of SG applications like SOFTprotection and FLISR. In a first step, it is possible to model the desired application using Function Blocks (FBs), as described in IEC 61499 [C+10]. Afterward, using a 'match-making' process, it should be possible to map the functionally-described application to a real device infrastructure, i.e., an existing SG, given constraints such as timing and safety. Then, the aforementioned system should propose different solutions that match the given constraints and let the user choose one among them. The chosen solution can then either be further adapted or directly deployed to devices in the field.

For describing the application functionally using FBs from IEC 61499, a 'database' (library) is needed. This database should model possible functionalities in the SG domain. For the mapping of the FBs to the real infrastructure, there needs to be a link between the manufacturer database and the function database. This link relates devices to the functions they support. Hence, it allows mapping and deploying the functionally described application to a real device infrastructure under given constraints.

To describe the given infrastructure, some method has to be chosen or developed. Here, the already existing standard IEC 61850 [C+12b] could be used. This standard not only allows the description of the infrastructure but also the use of 'generic' devices. Of course, it would also be possible to use other concepts like PSAL [PA], but it makes more sense to rely on widely used standards. These standards allow the reuse of existing tools and toolchains and guarantee interoperability. In particular, IEC 61850 offers concepts to create a manufacturer database of devices and their metadata in a standardized format. Metadata in this context refers to all information that could be used as constraints such as failure rate and latency. This manufacturer database could then be used to link specific devices to the generic devices used before. Of course, when describing the existing infrastructure, it should also be possible to use the real already existing devices.

The framework to be developed shall be able to decide whether the given application can

[1] https://www.iec.ch/smartgrid/roadmap/

be implemented on the available infrastructure or not. If so, it should propose which FBs shall be executed on which device and deployment files should be generated. Otherwise, it shall report an error. To easily develop such an application, some kind of toolchain has to be created that allows creating the functional description by selecting, placing, and connecting the FBs, selecting the important constraints, loading the structure of the existing SG, and proposing possible solutions.

1.3 Methodology

The methodological approach to reach the expected result comprises the following steps:

1. **Literature Review:** As a first step, relevant literature and information have to be gathered such as standards in the SG domain, similar existing frameworks in other domains as well as already existing approaches and background information. Furthermore, the theoretical background as well as the practical procedure of creating SG applications and working with existing SG standards have to be studied in depth.

2. **Application Description:** After studying the literature, possible functions of devices in the SG domain have to be identified, to then integrate them in a flexible, extendable database. Furthermore, it is necessary to examine how these functions can be represented as generically and flexible as possible using FBs. These FBs will then be connected to describe desired applications. Ideally, this step will not require any knowledge about the infrastructure. Because this task is similar to work done in project Auteg [DPK09], some parts could be reused and investigated.

3. **Infrastructure Description:** This step comprises modeling the existing infrastructure. Therefore, the already existing standard IEC 61850 can be used. This standard not only provides possibilities for describing generic as well as specific devices but also allows the description of the ICT infrastructure. Furthermore, it has to be investigated how metadata can best be integrated and how the manufacturer database can be built. It is also important to establish links between the database and the devices actually used.

4. **Mapping:** After it is possible to describe the functionality as well as the existing infrastructure, a mapping algorithm has to be developed, which will propose different solutions on how to map the application to real devices under given constraints. Furthermore, this algorithm should detect whether it is possible to implement the given function or not.

5. **Deployment:** Here, two different concepts have to be considered. Firstly, the IEC 61499 application must be deployed and secondly, the IEC 61850 devices must be configured. Both standards already offer concepts for this. IEC 61499 requires a runtime environment on the devices. Here it is necessary to find out which runtime

environments and communication protocols would be suitable. IEC 61850 provides a different concept. The runtime environment is already built into the device. These devices can then be configured using the services provided, for example by the use of standardized XML files.

6. **Toolchain:** Furthermore, to support the user in creating applications and using the proposed framework, a complete toolchain is proposed. This supports the user from the modelling to the deployment. Since this is not the main focus of this thesis, existing solutions should be used whenever possible to keep the engineering effort for this task as low as possible.

7. **Proof of Concept:** In a final step, the developed framework is used to model the overcurrent detection use-case, which corresponds to the first step of the FLISR scenario. If an overcurrent is detected, this should be logged and the corresponding technician should be informed. The created application should then be applied to a small testbed and generate corresponding deployments.

8. **Outlook:** Finally, possible extensions and modifications to the framework are discussed. Questions like, how to integrate more constraints, how to improve the search for optimal solutions, but also how to improve the mapping algorithm itself are discussed there.

1.4 Structure of this Thesis

In accordance with the methodology, the thesis is structured as follows: Chapter 2 presents the state of the art of SG related technology and standards as well as related scientific work and similar solutions in other domains. Chapter 3 then examines requirements and discusses different possibilities to create a framework for modelling and automatically deploying applications in the SG domain. In Chapter 4, which can be considered the main part of this thesis, this framework is elaborated in detail. Chapter 5 then applies the framework on parts of the FLISR scenario. The last chapter summarizes the thesis and gives an outlook on possible adaptations for the future.

CHAPTER 2

State of the Art

This chapter describes the standards and tools currently used to model, implement, and deploy SG applications. Furthermore, some scientific publications that support this process are also described here.

2.1 Smart Grid Models

In the introduction, the reasons for the establishment of SGs and the accompanying advantages and disadvantages have already been discussed. This leads to the question of what a SG looks like and which components it typically contains.

2.1.1 Smart Grid Definition

According to Stuart Borlase [Bor16], from a high-level system perspective the SG contains the following four major components:

Smart Sensing and Metering Technologies: provide faster and more accurate responses, in order to enable remote monitoring, time of use pricing, and demand-side management.

Integrated, Standard-Based, Two-Way Communication Infrastructure: provides an open architecture that enables both real-time information and control of every endpoint.

Advanced Control Methods: enable the monitoring of critical components, as well as diagnosis and troubleshooting.

Software System Architecture: supports the user in the best possible way in the decision-making process through improved interfaces, analytics, and visualization.

The SG can also be seen as a "framework for solutions" [Bor16]. Borlase divides this solution framework into three levels: the electricity infrastructure level, the smart infrastructure level, and the SG solution level, see Table 2.1. On each of these levels, there can be different SG applications, which must be able to interoperate horizontally as well as vertically. [Bor16]

Smart gird solutions	Utility Enterprise Applications				
	Operational efficiency	Reliability and security	Energy efficiency	Alternative energy	Consumer participation
Smart infrastructure	Engineering and Operational Systems				
	Communications Infrastructure				
	Smart Sensors, controllers, and meters				
Electrical infrastructure	Transmission and distribution infrastructure				
	Alternative energy sources, storage, and EVs				
	Energy consumer home area network				

Table 2.1: Smart grid technology framework [Bor16]

2.1.2 Smart Grid Architecture Model

As SG technologies can span the entire electric grid, there must be some kind of interoperability between different components. This can be achieved by using standards. For example, the IEC is working on the standardization of various areas of SGs, see also the IEC Smart Grid Roadmap[1]. The Smart Grid Architecture Model (SGAM) was developed as a result of the EU's efforts to identify gaps in the SG standardization process. This was done within the scope of the M/490 Standardization Mandate to the European committees CEN, CENELEC and ETSI. However, it turned out that the framework is also well suited for describing SG use-cases in general. [CCE12] SGAM is a use-case based methodology, which is based on a three-dimensional framework as can be seen in Figure 2.1.

This framework is divided along the z-axis into planes, also called interoperability layers. Layers are used to represent use-cases from different perspectives [CCE12][WMS17][PA] [GUD17]:

- **Business Layer**: This layer represents the business view of the information exchange in the context of SGs. Thus, regulatory and economic structures and policies can be represented with the help of this layer.
- **Function Layer**: This layer describes the interaction of functions and services independent of actuators and their physical implementations.
- **Information Layer**: This layer describes the information exchange between functions, services, and components. It also includes the semantics and structure of the exchanged data.

[1] https://www.iec.ch/smartgrid/roadmap/

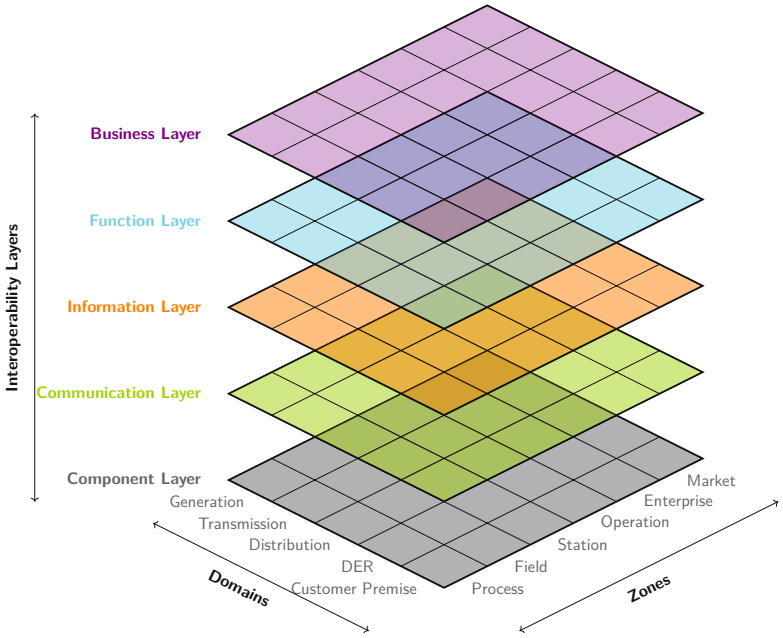


Figure 2.1: SGAM - Smart Grid Architecture Model [PA]

- **Communication Layer**: Here, the used protocols and mechanisms are described, which guarantee the interoperable exchange of information.
- **Component Layer**: This layer finally describes the physical distribution of all existing components, such as actors and other equipment.

These layers are further subdivided along the x- and y-axis into domains and zones. The domains describe the 'physical' where in terms of the energy supply domain [CCE12][WMS17] [PA][GUD17]:

- **Bulk Generation**: This domain includes large scale power plants, ranging from nuclear to renewable hydro power.
- **Transmission**: In this domain, the transport of generated energy over great distances takes place.
- **Distribution**: Here, the transported energy is distributed to and from resources.
- **Distributed Energy Resources**: This domain includes energy resources that produce, store, or consume energy.
- **Customer Premises**: Customers ranging from industry to companies to private households are subsumed within this domain.

The zones describe the 'logical' where in terms of the automation pyramid, i.e., the hierarchical levels of power system management according to IEC 62357 - Power systems management and associated information exchange [C+16][CCE12][WMS17][PA][GUD17]:

- **Market**: In the Market zone, operations possible along the energy conversion chain like energy trading are covered.
- **Enterprise**: Here, commercial and organizational processes, services, and infrastructures are covered.
- **Operation**: This zone covers power system control operations like Distribution Management Systems (DMS) and Energy Management System (EMS).
- **Station**: Here, areal aggregation takes place.
- **Field**: This zone covers equipment to protect, control, and monitor the process of the power system.
- **Process**: This zone covers the physical, chemical, or spatial transformations of energy, as well as the physical equipment directly involved.

SGAM not only ensures a structured view of the entire SG system but also allows individual interoperability aspects to be considered. Furthermore, it allows the validation of SG use-cases and their support by standards. In order to architect SG systems according to the SGAM, Christian Neureither proposed a "Model Driven Engineering Approach for Systems Engineering in the Smart Grid" called SGAM Toolbox [Neu17].

2.2 Use-Case Design and System Description

When developing a new SG application, the first step is the planning and design phase. The new use-case has to be described and specified first. In the case of a SG application, the existing network topology has to be modeled and the desired behavior and architecture have to be specified. In the following, existing state-of-the-art methods are described that enable this process.

2.2.1 IEC 62559 - Use-Case Methodology

Due to the increasing complexity of power system automation, Electric Power Research Institute (EPRI) developed a new approach in the early 2000s. The idea was to use a system engineering approach, which allows a structured implementation. Since then, this approach has become a separate IEC standard, specifically IEC 62559. It also represents one of the most widely used approaches for the description of SG use-cases. Nevertheless, this method is not limited to the SG domain, but can also be used in many other domains. This approach is also used in the PoSyCo project to describe the use-cases [oT15]. [C+15][PA][GUD17]

IEC 62559 provides a standardized use-case template for the description of new applications. In addition to the step-by-step analysis of the use-cases, this includes the different actors, the information to be exchanged, and various requirements. Part 3 of the standard also describes how (de-)serialization to eXtensible Markup Language (XML) works. This allows exchanging the model between tools of different verndors. [C+15]

Although this approach is often used, there are also a few drawbacks. Its output is still a text document. Therefore, collaborative development is difficult. Furthermore, the use-case description is rather for documentation purposes. This makes automatic code generation very difficult or even impossible. [PA]

2.2.2 CIM - Common Information Model

Due to the increasing deregulation of the energy market, it became more and more important to simplify the data exchange between different energy companies. This exchange is necessary to guarantee the reliable operation of interconnected power networks. The traditional use of proprietary tools and protocols by different companies would lead to an exponential increase in complexity as more and more different applications would be integrated. This was one of the main drivers for developing the Common Information Model (CIM). [McM07][PA]

The origins of CIM date back to the early 1990s. CIM was created within the framework of an EPRI project as an internal data model for a common EMS. Over time, it was adapted and extended several times by major EMS vendors. Thus, it became an IEC standard for data exchange between applications in the energy sector. More precisely, CIM currently consists of three different parts: IEC 69170, IEC 61968, and IEC 62325. While IEC 61970 focuses on power transmission systems, particularly their EMS, IEC 61968 and IEC 62325 extend CIM with data for DMSs and messages for market communication, respectively. [C+08][McM07][PA]

CIM itself is a powerful, object-oriented data model that also provides interfaces for power system applications. Additionally, it offers a comprehensive domain ontology for the energy sector. Since CIM is maintained in Unified Modelling Language (UML), it is also platform and technology independent. Even though the data models are maintained in UML, the standard describes how a mapping to the Resource Description Framework (RDF) could look like by providing an XML Schema. RDF in this context is needed for expressing the relation between components. This enables the serialization of generated models and topologies. [C+08]

Figure 2.2 illustrates a section of the UML class diagram of CIM. Among other things, it contains a (circuit) breaker. This breaker inherits and extends the functionality of a switch. Figure 2.3 illustrates the use of these classes by providing a small topology example. Individual instances, such as a breaker, are connected via their terminals using connectivity nodes. In this way, even large systems can be represented in detail. [McM07][C+08]

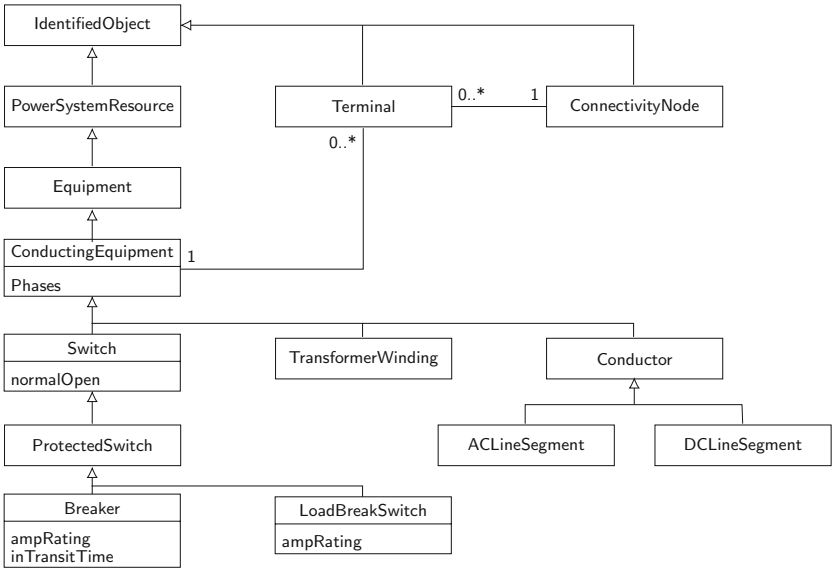


Figure 2.2: Excerpt of the CIM class diagram [McM07]

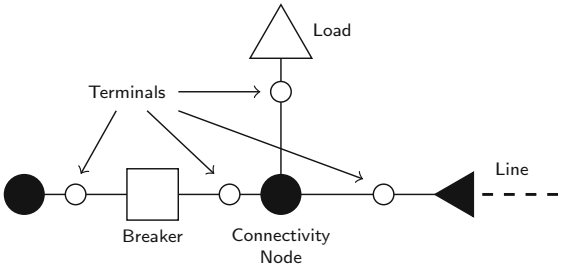


Figure 2.3: Graphical representation of a CIM model [McM07]

Even though CIM is widely used for modeling power grid topologies, it is neither possible nor intended to model automation and control functionalities. For this, it would be necessary to combine CIM with other approaches. [PA]

2.3 Power Utility Automation and Application Modeling

A SG consists of many different, mainly distributed devices that must communicate with each other. This section deals with methods for modeling and describing applications in this complex, distributed environment.

2.3.1 IEC 61131 - Programmable Controllers

IEC 61131, named "Programmable Controllers", has been first published in the 1990s. Its goal has been the standardization of the architecture of Programmable Logic Con-

trollers (PLCs). PLCs are small industrial computers used in the automation industry. Furthermore, the standard provides approaches to programming these devices. While today there is hardly any vendor that does not support at least parts of IEC 61131, before the standard has been published each vendor had its own architecture. As a consequence, devices from different vendors were not compatible. But still, there are many vendor-specific extensions, which counteracts the idea of interoperability. [C+12a][ZL14]

IEC 61131 consists of nine parts, where part three is probably the most prominent one. This part not only defines programming languages but also common elements like data types and information exchange for all of those languages. Even a standard function library is defined inIEC 61131-3. The supported languages are: [C+12a]

- Instruction List (IL), textual, deprecated
- Structured Text (ST), textual
- Function Block Diagram (FBD), graphical
- Ladder Diagram (LD), graphical
- Sequential Function Chart (SFC), graphical

In general, IEC 61131 applications are executed using a cyclic system model. At the beginning of each cycle, the process inputs are read and provided to the program. The program then calculates the outputs, which are written at the end of the cycle. Furthermore, SFCs provide elements for sequential and parallel execution. [C+12a]

Especially the graphical languages such as the FBD have a high-level abstraction compared to 'conventional' programming languages to keep it readable even for non-computer technicians. Since those languages are automatically convertible, the rest of this section focuses on FBDs. Here, functions are represented using so-called function blocks as can be seen in Figure 2.4. [C+12a]

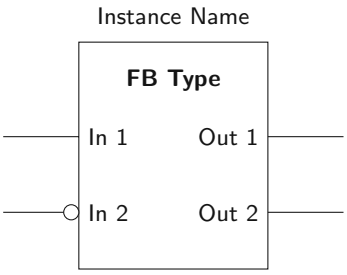


Figure 2.4: IEC 61131 - Function block

The inputs are on the left side of the 'block' and the outputs are on the right side. Furthermore, a boolean value can be negated, which is represented by a small circle

next to the input. An FB type is uniquely identified by its type name. In a program, it is of course possible to instantiate more than one block of a type. Those instances are then uniquely identified by their instance names written on top of the block. Each input and output has some data type – may it be a basic type like integer or a composite datatype – assigned. Assuming corresponding inputs and outputs are of compatible types, it is possible to connect them to create a whole application as illustrated in Figure 2.5. [C+12a]

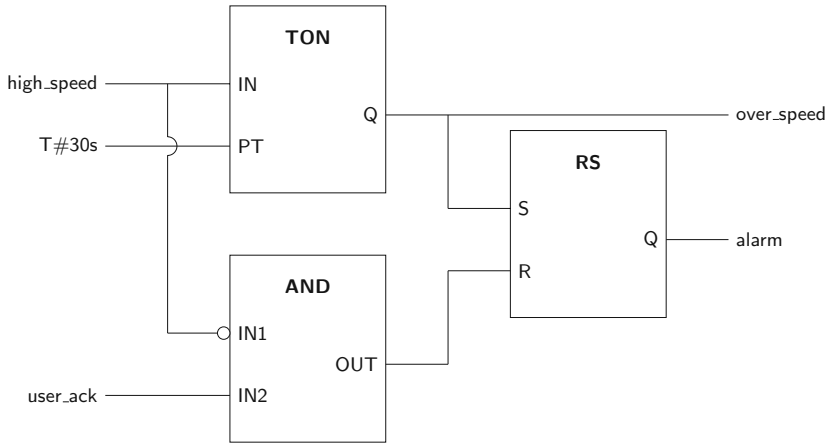


Figure 2.5: IEC 61131 - FBD example application[2]

This example application consists of three different FBs already defined in the standard library: Timer On Delay (TON), AND and Reset Set (RS). It represents an application for some agitator for example, which can only be used in high-speed mode for a limited period of time. In this example, after 30 s in high-speed mode, an over-speed signal and an alarm will be triggered. The alarm will only mute when high speed is turned off and a user acknowledges the alarm. [C+12a]

2.3.2 IEC 61499 - Distributed Control Reference Model

Even though IEC 61131 provides good concepts for handling centralized control applications, the industry has since then moved to a more decentralized approach. In contrast to a centralized system, where a central intelligence is controlling all involved components, a decentralized system consists of a vast number of individual parts. These individual subsystems have their own intelligence and communicate with each other to solve the given tasks as a whole. Attempting to apply IEC 61131 to decentralized systems results in several problems. For example, IEC 61131 doesn't provide a complete view on distributed application and the communication between individual PLCs is not standardized within IEC 61131. Furthermore, the cyclic execution model of IEC 61131 is not well suited for distributed applications since feedbacks depend on the underlying

[2] https://www.eclipse.org/4diac/en_help.php?helppage=html/before4DIAC/iec61499.html

software implementation. Hence, an event-based approach would be desirable. This triggered the IEC to develop a new standard. [C+10][PA][ZL14]

In the early 2000s, the IEC published the standard IEC 61499, the distributed control reference model. Besides interoperability, other objectives of the IEC were configurability and portability. Portability in this context ensures the compatibility of FBs and devices from different vendors. It further guarantees a uniform way to configure them. The use of IEC 61499 to implement automation functions for SGs has already been suggested in several publications and was proposed in the German Smart Grids Standardization Roadmap as well as in the IEC Smart Grid Standardization Roadmap. [C+10][PA][ZL14][(SG10]

Execution Model

The modeling language used in IEC 61499 is similar to IEC 61131 FBDs shown in the previous section. However, it is designed for decentralized systems and allows to model complete applications rather than programming individual PLCs. The standard intentionally does not support global variables or global states. Instead, all functionality is encapsulated in FBs. This approach simplifies the deployment of FBs accross multiple PLCs. Furthermore, the cyclic execution model of the FBs from IEC 61131 is replaced by an event-driven execution model. This standardizes the communication between blocks. An application is thus modeled by connecting individual FBs. The standard also offers the possibilities to represent devices, their connections, and the mapping of the function blocks to the individual devices. [C+12a][C+10][ZL14]

As with the IEC 61131 FBs, an IEC 61499 FB (see Figure 2.6) encapsulates the corresponding functionality. Again, the inputs are located on the left, whereas the outputs are located on the right. The interface is divided into event and data inputs and outputs. The events are located above the block name and the data below. Both events and data have a type compatible with those of IEC 61131. Accordingly, only inputs and outputs that have the same data type can be connected. However, data inputs must be unambiguous, i.e., several outputs cannot be connected to one input. With events, this is indeed possible. Moreover, vertical lines between event inputs and data inputs, or event outputs and data outputs, respectively, indicate which data inputs are read at which event input and which data output are written at which output event. This link is symbolized by small squares, which are connected by lines. An event triggers the reading of the connected input data as well as the execution of the FB.

Function Block Types

In the simplest case, the algorithm to be executed is determined by so-called Electronic Control Charts (ECCs). In principle, an ECC is a state machine. The algorithms associated with each state can be written, for example, in a language defined in IEC 61131-3. But also other languages might be feasible. [C+10][ZL14]

Figure 2.7 shows an example of an ECC. The state machine starts in the initial state. If an event occurs and there is a corresponding outgoing edge from the current state,

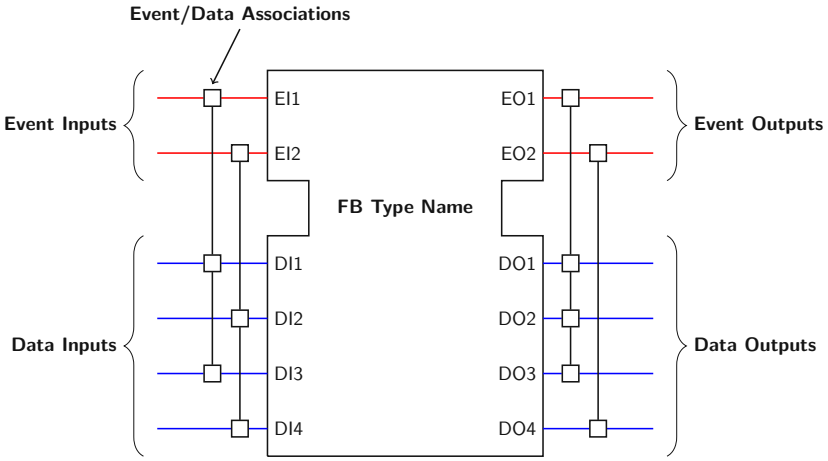


Figure 2.6: IEC 61499 Function block

the state changes along the edge. A transition may not only depend on a specific event, but it is also possible to define additional conditions. In the example depicted, one can see that a transition from State 1 to State 2 is only possible after the input event EI 2 occurs and the internal variable $V1$ equals 5. The additional condition is put after the event in square brackets. After a transition, all algorithms belonging to this new state are executed one after the other and the corresponding output events are fired. Subsequently, the system waits for the next event. Edges that are marked with a 1 are an exception; those are taken immediately. The FBs just described are also called Basic Function Blocks (BFBs), see Figure 2.8a.

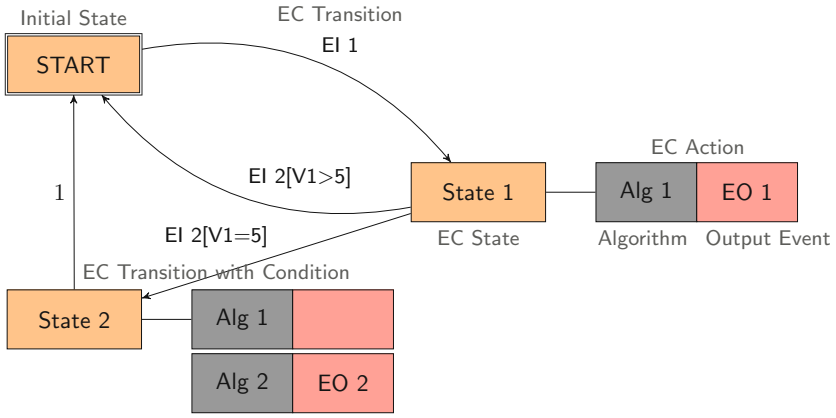


Figure 2.7: IEC 61499 Execution Control Chart

Besides these BFBs, which use ECCs to describe their functionality, there are also other types of function blocks: [C+10][ZL14]

- Composite Function Blocks (CFBs) (see Figure 2.8b): A CFB is a function block

whose functionality is described by a network of other function blocks. This network is connected to the inputs and outputs of the CFB.

- Subapplications (see Figure 2.8c): Like a CFB, a subapplication also contains a network of FBs. But in contrast to a CFB, a subapplication acts like a container to improve visibility. A subapplication itself does not sample inputs but these are passed on directly to the FBs inside. This implies that subapplications can be distributed to multiple devices, unlike CFBs.

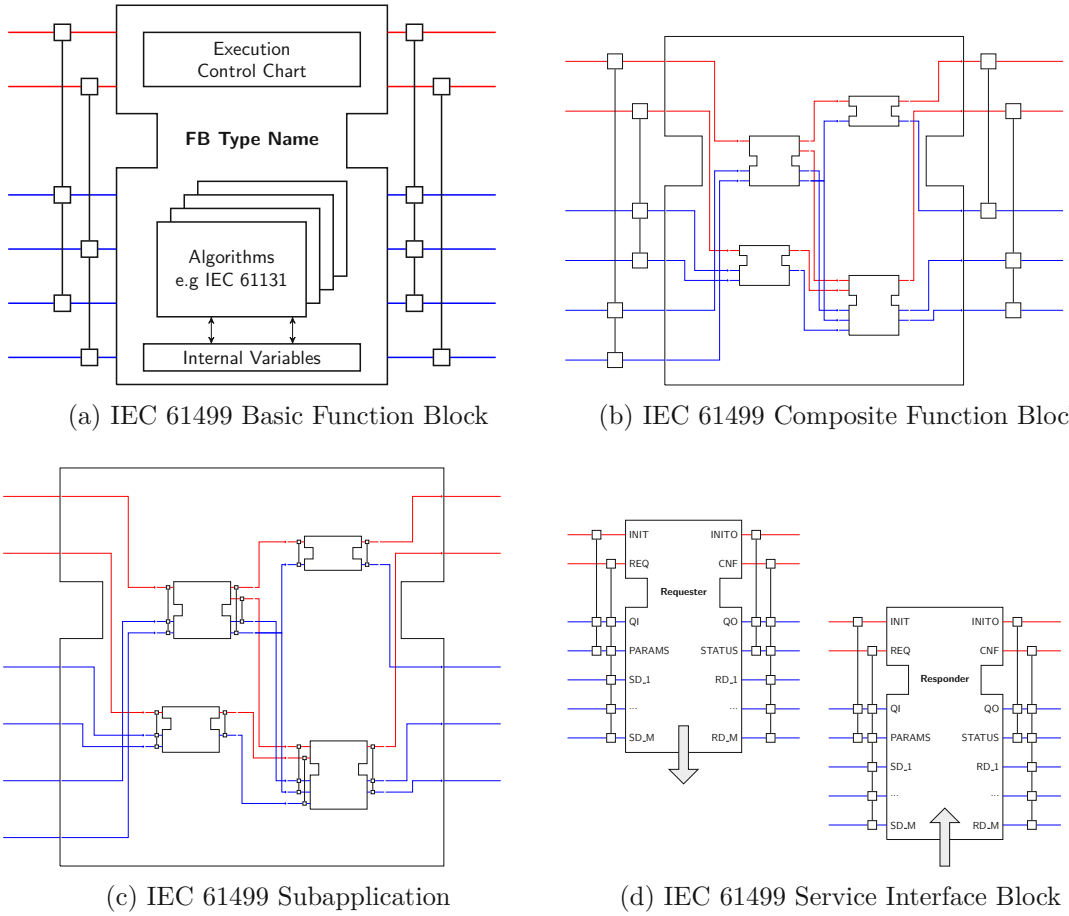
- Service Interface Function Blocks (SIFBs) (see Figure 2.8d): These FBs are used to encapsulate communication services and to access specific parts of the hardware. They only describe a service using sequence diagrams, while the source code is hidden and cannot be accessed. There are two types of SIFBs: a requester FB, which requests some service(s) from the underlying system, and a responder FB that can react to events of the underlying system. Figure 2.9 shows two examples of sequence diagrams. While Figure 2.9a shows how a SIFB could be used to model a client-server communication, Figure 2.9b shows this for a publish-subscribe communication. These sequence diagrams are independent of the underlying communication protocol. Unlike the other types of FBs, SIFBs cannot only be activated by incoming events, but also by the hardware. Thus, a responder FB can trigger an event every time a packet arrives.

Furthermore, Adapter Inferfaces enable using a compact view for a certain subset of event and data flows. Adapter Interfaces allow to define a combination of event and data I/Os, which can then be used either as an adapter provider, i.e., a plug, or as an adapter acceptor, i.e., a socket. Sockets and plugs always have a mirrored interface. With this concept, it is possible to reduce or structure the I/Os of the provider and acceptor. Figure 2.10 shows a small example. As with SIFBs, adapters are described using sequence diagrams. [C+10][HSM+12]

Application Deployment

As stated above, the standard not only allows modeling individual applications but also modeling entire systems. This includes the devices, the communication infrastructure, and the controlled process/machine. Using this model, the FBs of the applications can now be distributed to the individual devices or even to the different resources of these devices. To start, stop, and deploy FBs, management commands defined by the standard can be utilized. For the distributed communication, SIFBs are generated in the system model, which makes the data and events available to the function blocks connected in the application model via the communication interface. This is illustrated in Figure 2.11. [C+10][ZL14]

Finally, the standard also defines so-called Compliance Profiles. Because some aspects are deliberately left open in the standard to ensure the greatest possible flexibility, it is only



(a) IEC 61499 Basic Function Block

(b) IEC 61499 Composite Function Block

(c) IEC 61499 Subapplication

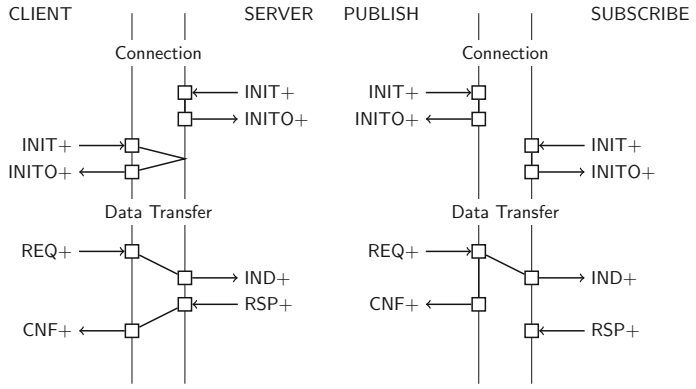
(d) IEC 61499 Service Interface Block

Figure 2.8: IEC 61499 Function Block Types

described how these aspects must be specified. This specification is called Compliance Profile and serves to improve interoperability between different vendors. [C+10]

2.3.3 IEC 61850 - Communication Networks and Systems

The IEC 61850 standard was first published between 2003 and 2005 under the name "Communication networks and systems in substations". Like CIM, it is maintained by the IEC Technical Committee (TC) 57. The original aim of this standard was to achieve uniform communication within a substation. Since its first publication, however, it has been extended several times to also include automation outside of substations. Accordingly, the standard has been renamed to "Communication networks and systems for power utility automation". The underlying idea is that information to be exchanged between devices and systems is standardized and made available in an object-oriented manner. This uniform representation guarantees interoperability, while the concrete implementation of the functionality remains in the hands of the vendors. [BBBW03][Mac07][C+12b]



(a) Client - Server (b) Publish - Subscribe

Figure 2.9: IEC 61499 Sequence Diagrams for Client-Server and Publish-Subscribe communication patterns [PA]

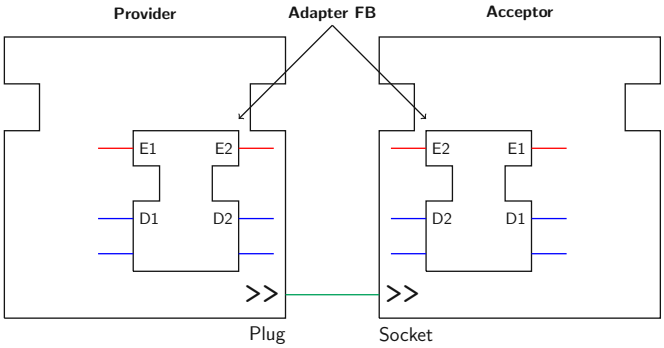


Figure 2.10: IEC 61499 Adapter Interface

The standard consists of ten parts, which are supplemented by numerous Technical Reports (TRs) and specifications. Thematically, the individual parts can be divided into the subject areas of modeling, communication, and configuration. Furthermore, the standard provides a conformance testing methodology to inspect implementations. [BBBW03][Mac07][PA][C+12b]

Modeling

The first parts of the standard are dedicated to the identification of the diverse requirements of a substation. Based on these requirements, the necessary services and data models ,which are capable of coping with them, were defined . This approach has the advantage that the model and the services can later be mapped to any protocol. [C+12b]

IEC 61850 defines a hierarchical, object-oriented data model as can be seen in Figure 2.12a. This hierarchical structure allows accessing objects using self-describing strings, see Figure 2.12b. The model starts with a Physical Device (PD), also called Intelligent

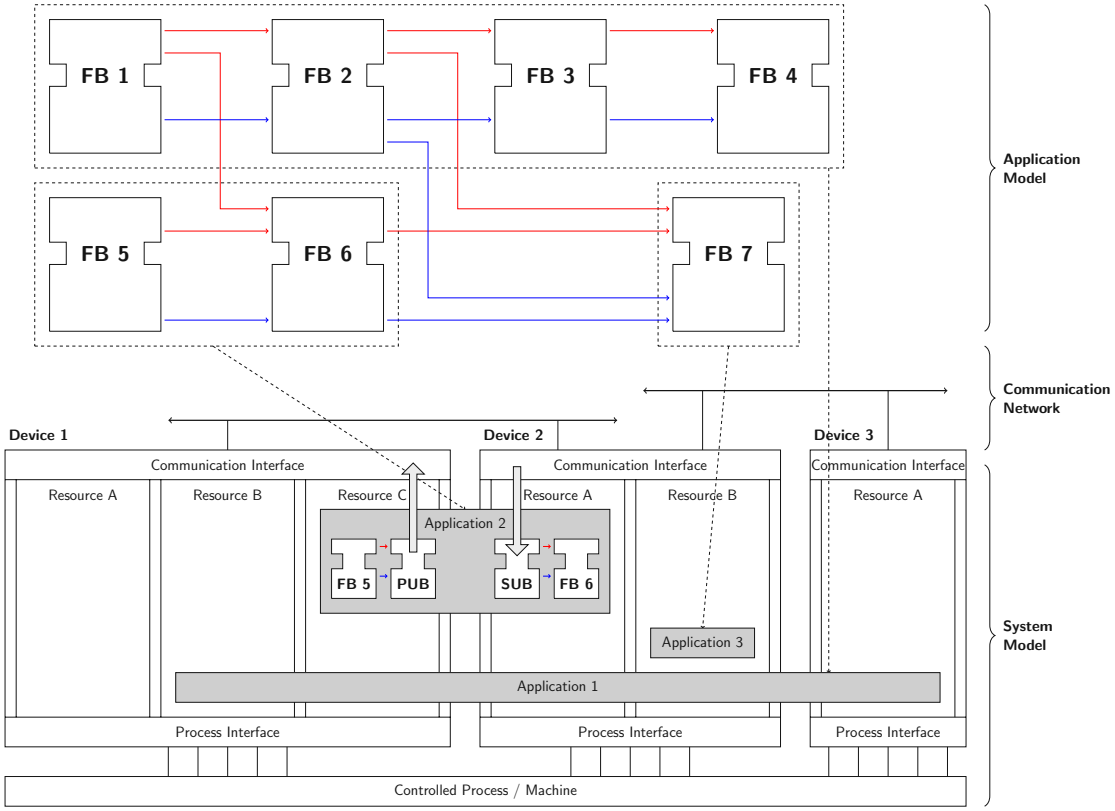
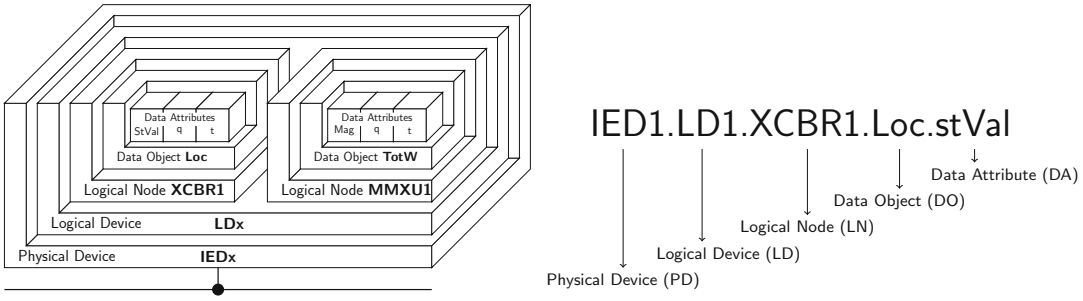


Figure 2.11: IEC 61499 Deployment of a Distributed Application [PA]

Electronic Device (IED). An IED is a device that is connected to a network and can be identified by its network address. Such a PD can contain one or more LDs. This allows the PD to act as a proxy for the individual LDs. LDs can contain several Logical Nodes (LNs). [BW02] [BBBW03] [Mac07] [C+12b]



(a) Data Model Hierarchy [SFA18] (b) Self-Describing String [Mac07]

Figure 2.12: IEC 61850 - Data Model Hierarchy Example [Mac07]

So-called LNs form the smallest functional unit of this model and represent some power

system function. They are groupings of data and associated services. The standard describes several different types of LNs, such as for measuring, starting with the letter 'M' and for controlling and monitoring, starting with the letter 'C'. Furthermore, special LNs are used for the description of the LD (LLN0) and the PD (LPHD). As mentioned, LNs contain several objects. The concept of Common Data Classes (CDCs) was developed for this purpose. These are general blocks that bundle related data into one type. Examples would be the type Single Point Status (SPS) for a binary readable status value or Double Point Control (DPC) for a settable control value with four possible set points. [Mac07] [C+12b]

Table 2.2 illustrates this concept. It shows the structure of a circuit breaker, which is modeled as an XCBR-LN. Among other services and attributes that are common to all LNs, it has the mandatory attribute 'Loc', which describes whether or not it is a local operation. The type, i.e., the CDC, of this attribute is 'Single Point Status', i.e., a readable binary value. Table 2.3 shows which attributes an object of type SPS has to contain, categorized by their attribute types (Functional Constraint). Additionally, the name, the data type, the value range, and triggers that trigger a change are also described. Each attribute is either Mandatory (M) or Optional (O). The standard uses this shown tabular format to describe all kinds of CDCs and LNs. [Mac07][C+12b]

XCBR				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Loc	SPS	Local operation (local means without substation automation communication, hardware direct control)		M
EEHealth	INS	External equipment health		O
EEName	DPL	External equipment name plate		O
OpCnt	INS	Operating counter		M
Controls				
Pos	DPC	Switch position		M
BlkOpn	SPC	Block opening		M
BlkCls	SPC	Block closing		M
ChaMotEna	SPC	Charger motor enabled		O
Metered Values				
SumSwARs	BCR	Sum of switched Amperes, resetable		O
Status Information				
CBOpCap	INS	Circuit breaker operation capability		M
POWCap	INS	Point on Wave switching capability		O
MaxOpCap	INS	Circuit breaker operating capability when fully charged		O

Table 2.2: IEC 61850 XCBR Logical Node [C+12b]

In addition to the abstract data model, the standard also describes a set of services and the responses to them, also called the Abstract Communication Service Interface (ASCI). This allows all IEDs to behave identically in the network. The models of ASCI include services for reading, writing, controlling, event-driven reporting, and file transfer. The independence between abstract communication service and communication technology can sustain this standard for a long time. [BW02] [C+12b]

SPS - Single Point Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
stVal	BOOLEAN	ST	dchg, dupt	TRUE | FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
units	Unit	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 2.3: IEC 61850 SPS Common Data Class [C+12b]

Communication

As already mentioned, the standard intentionally defines abstract data models and services. Thus, these can theoretically be mapped to any protocol. Part 8-1 of the standard describes the mapping to the Manufacturing Messaging Specification (MMS) protocol. This is an international packet standard (IEC 9506), which is based on a client-server principle. Due to its versatility, it easily supports the complex name and service models of IEC 61850. MMS already offers a wide range of flexible services. For example, the ASCI services 'GetDataValues' and 'SetDataValues' can be mapped to the MMS services 'read' and 'write' respectively. But MMS also supports services for creating and editing files and many more. LDs are mapped to MMS domains, LNs to MMS named variables. [C+12b] [Mac07]

Besides these mappings, the standard also describes profiles for supporting both real-time interaction and information monitoring between smart devices, which are dependent on the respective services. Among other things, Generic Object Oriented Substation Event (GOOSE) models are provided, which enable fast and reliable transmission of data. This is a publish-subscribe based protocol, which packages the data directly in the Ethernet link layer and thus dispenses with TCP/IP. GOOSE allows any format of data to be grouped into a data set and transmitted within a period of 4 milliseconds. Generic Substation Status Events (GSSE), an extension of GOOSE – where only status data can be exchanged, time synchronization via Simple Network Time Protocol (SNTP), and the distribution of Sampled Measured Value (SMV) via multicast are also supported. Figure 2.13 gives an overview of the communication profiles used in IEC 61850. However, it should be noted

that the configuration on how the IEDs exchange their data is done via MMS. This is also the protocol that Supervisory Control and Data Acquisition (SCADA) systems are typically connected to, hence the vertical communication. Horizontal communication is mostly done via GOOSE and GSSE. [Hua18] [PA] [Mac07] [C+12b]

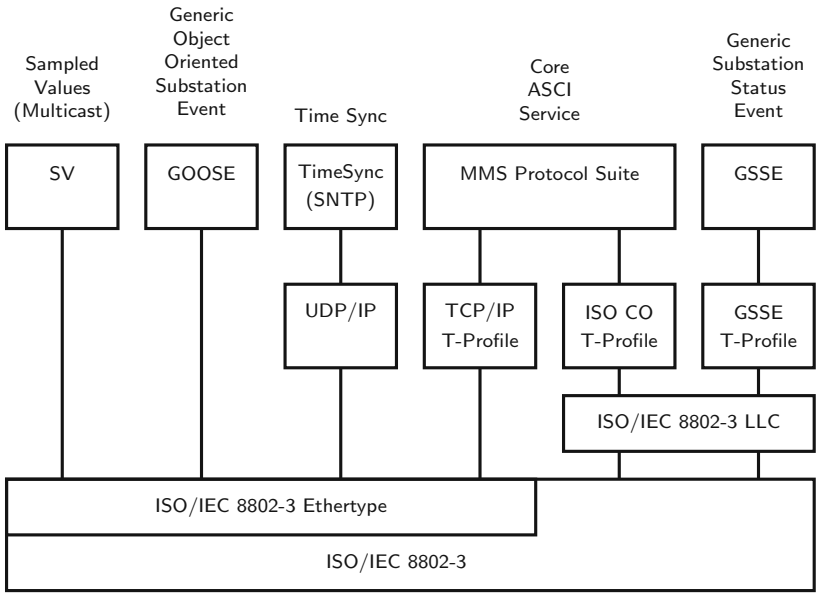


Figure 2.13: IEC 61850 Communication Profiles

Besides, the standard allows the digitization of the base values at the source and the transmission of these sampled values to the station. This is realized via the so-called Merging Units (MUs). SMVs, which are potentially sampled via a proprietary connection, are then sent to the corresponding IEDs via the process bus by multicast. Access to status information and control operations is also possible. [Mac07] [C+12b]

Configuration

Part 6 of the standard defines an XML-based configuration language for stations, called System Configuration Language (SCL). It consists of a hierarchy of configuration files that allow different levels of the overall system to be described in unique and standardized XML formats. [Hua18] [C+12b]

These include:

- **System Specification Description (SSD):** A file, which is used to describe both the topology of the electrical system (as a single line diagram), as well as the required LNs without including the specific IEDs, which implement these LNs.
- **IED Capabilities Description (ICD):** This is a file provided by vendors, describing the capabilities of their devices, i.e., which LNs and which optional object-

s/attributes are supported. It acts as a template in a system or IED configuration tool.

- **Substation Configuration Description (SCD):** Here, the complete configuration of the system including the information model of the real equipment as well as network parameters, data flow, and the relationship with the electrical topology is described.
- **Configured IED Description (CID):** This file is meant to be sent directly to one specific IED for configuration. It includes the information published in GOOSE messages and the information available for the different clients. An Configured IED Description (CID) is similar to an Substation Configuration Description (SCD) but only contains the information relevant for one specific device.
- **Instantiated IED Description (IID):** This file contains a pre-configured instance for a specific IED, which can be imported to a system configuration tool.
- **System Exchange Description (SED):** This file serves the description of interfaces for data exchange between different projects.

Each of these files can consist of up to five sections depending on its type:

- **Header:** identifies the version;
- **Substation:** lists and provides information about the different entities and their physical interconnection;
- **Communication:** describes the different communication points;
- **IED:** describes the complete configuration of IED(s);
- **DataTypeTemplates:** defines different instances of LNs, objects, attributes, and other details.

Figure 2.14 from top to bottom shows the whole engineering process including the usage of the different SCL files. Chapter 5 illustrates parts of the process through applying it to an specific use-case and shows how such an SCL file looks like.

2.4 Harmonization of Standards

Different standards use different views and data models, even if they operate within the same scope/domain. Harmonization attempts to reconcile these different perspectives in order to reduce both complexity and costs, such as for the more extensive compliance checks.

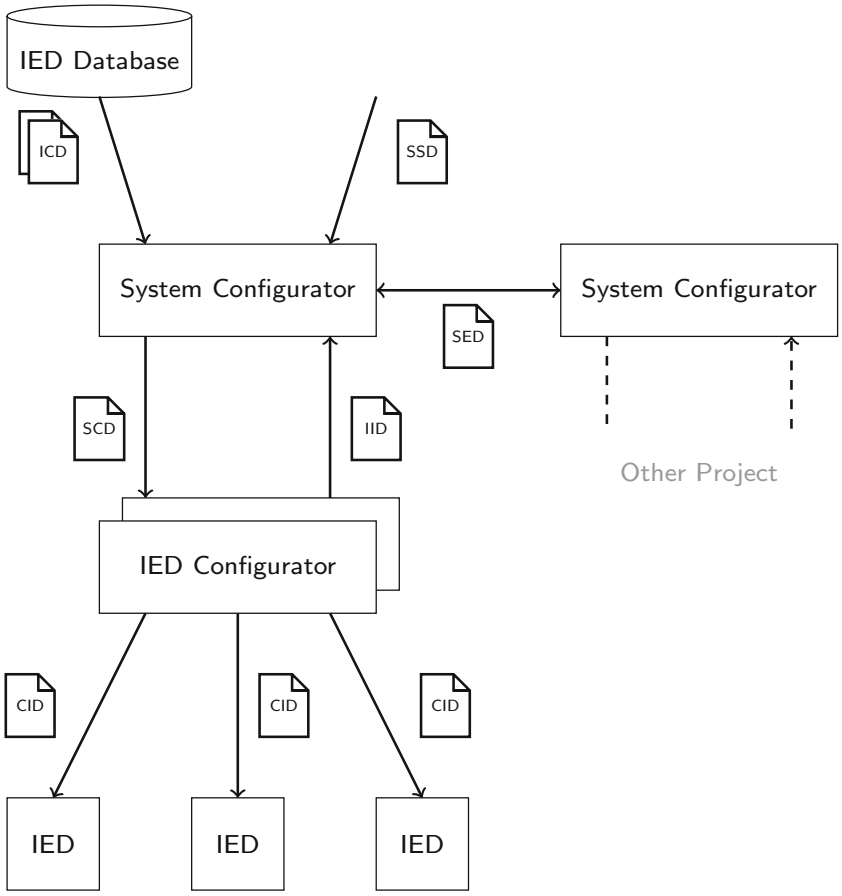


Figure 2.14: IEC 61850 Engineering Process [Wim14][KYL+11]

2.4.1 Harmonizing CIM and IEC 61850

Lately, much work had been done in order to harmonize CIM and IEC 61850 [BLS10] [MSS+12] [PSK+09] [SRS10] [Mac11]. The Working Group 19 of the TC 57 published a TR, which "describes a mapping for information exchange between power system installations based on the modeling approach of IEC 61850; and business systems based on IEC CIM standard data exchanges"[Ber19][C+18].

CIM and IEC 61850 satisfy different needs. When looking at the SGAM-plane, CIM operates from the Operation zone upwards, while IEC 61850 operates from the Station zone downwards (next to Companion Specification for Energy Metering (COSEM), which is used in the Customer Premises domain for Smart Metering communication). So while CIM has been designed for real-time information exchange via SCADA, IEC 61850's original aim has been the interoperable communication between intelligent devices. The idea behind this TR is to use an IEC 61850's SCD file and transform it into CIM-UML. Therefore a mapping has been provided. Furthermore, some recommendations have been outlined to reduce ambiguities. [Ber19][C+18]

2.4.2 Harmonizing IEC 61499 and IEC 61850

Towards a more "distributed intelligence rather than traditional centralized control" Vyatkin et al. presented a new approach to power system automation by combining IEC 61850 and IEC 61499 [HVNS11]. Their basic idea was to implement IEC 61850 LNs and logical devices using IEC 61499 basic and composite FBs. This guarantees a vendor-independent description of the LNs together with the functions that produce and consume the data objects of them and further allows easy simulation of the whole system. A similar approach has also been presented by Zhu et al. [ZSD11]. [HVNS11]

In [ZV12] they extended this approach to distributed multi-agent intelligence and introduced so-called intelligent Logical Nodes (iLNs). ILNs are CFBs that include the data of the corresponding LN. Besides, they also contain a service interpreter, which parses the name of the requested service and some intelligence, which is responsible for decision making and negotiation with other iLNs. This intelligence implements the algorithm of the iLN. In this approach, CDCs are modeled as structured data types. Figure 2.15 illustrates the concept for an XCBR-iLN.

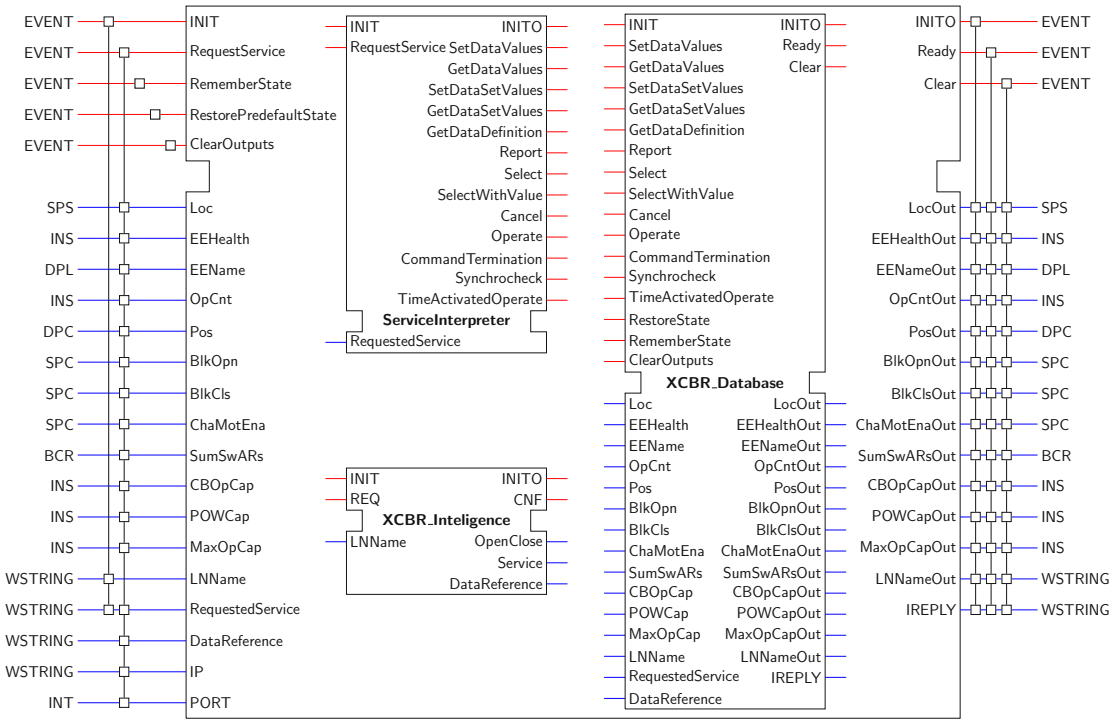


Figure 2.15: Combining IEC 61850 and IEC 61499 - Intelligent Logical Node [ZV12]

[YZV14] and [ZYV13] describe an IEC 61850/IEC 61499 based engineering process for SG automation. Here, Vyatkin et al. presents the so-called 'SysGridTool', which "ties the modeling process and the modeling language together" [YZV14]. The SysGridTool offers and adopted the IEC 61850 design process, where an FB network is automatically

generated from an SCD file. This network can then be used for testing and validation using co-simulation. More recently they have also shown an ontology-driven approach that reaches the same goal, but also allows additional information to be taken into account [YDV17].

In [dL19], Francisco de Lima proposes a new methodology where he enriches the substation specification from IEC 61850 with information about the internal model. This allows the automatic generation of an IEC 61499 system in order to be able to emulate the complete substation.

[VPT+14] shows how IEC 61499 can be used for describing control applications in IEC 61850 systems. Here, the internal logic of the IEDs is modeled using FBs, while for the communication with other LNs IEC 61850 is used.

2.5 Related Scientific Work of other Domains

Similar to the aim of this framework, the aim of the project Automatisierter Entwurf für die Gebäudeautomation (AUTEG) was to develop an automated approach for the design, construction, and commissioning of Building Automation (BA) systems. It was carried out between 2007 and 2009 by the TU Dresden in cooperation with the Helmut Schmidt University and partners from industry. After the requirements have been identified, a platform-neutral blueprint will be created automatically. After optional modifications, this can then be automatically converted into a platform-specific design with the help of a component database. This knowledge-based repository contains all components of the BA domain and enables the intelligent search for suitable, interoperable components. Furthermore, the idea is that this repository can be filled by the vendors themselves by providing standardized capability description files. [DPK10]

Besides AUTEG [DPK10], the successor projects Automatische Installation drahtloser Systeme der Gebäudeautomation (AUDRAGA) [PDRK13], Self-organising, Cooperative, and robUst Building Automation (SCUBA) [MRL+13] and Tools for Continuous Building Performance Auditing (TOPAS) [LLP+17] are based on the same process. These were also developed in cooperation with the TU Dresden and improve or extend the original toolchain, for example with wireless components or better selection algorithms. The work of Dibowski et al., who have comprehensively dealt with the semantic description of devices in the BA area, should also be emphasized here.

Based on the results of these research projects, the software Automatisierter Entwurf von Raumautomationssystemen (AUTERAS)[3] was developed. It combines the complete toolchain in a single program. It allows among other things to define requirements for different rooms and room types in a building, such as lighting, shading and Heating, Ventilation and Air-Conditioning (HVAC) needs. Based on these requirements and equipment already present, an RA-scheme according to Verein Deutscher Ingenieure (VDI) 3813 is generated automatically. This can now be edited manually. Besides, it

[3] https://auteras.de/

is possible to define non-functional device requirements such as the desired protection class for the generic function blocks. In a further step, an optimization process and the components repository described above are used to propose several solutions that fulfill both functional and non-functional requirements.

Another interesting concept has been proposed by Filip Pröstl. In his dissertation, he proposed a new rapid engineering methodology for the development of SG applications. For this purpose, Power System Automation Language (PSAL) has been developed. This language not only allows a SGAM compatibel description of use-cases, but also allows the "rapid development of automation, control, and ICT functions of power system applications" [ZAS18]. Furthermore, it enables automated transformation to IEC 61850-, IEC 61499- and SGAM-compliant models. [PA]

CHAPTER 3

Proposed Framework

The primary goal of the thesis is to develop a generic framework for application development in the SG domain. It aims to extend already existing SGs with additional measurement, control, and protection applications. It is expected that the application engineer will use IEC 61499 to describe the new applications. A device mapping function will then map the used function blocks to an already existing infrastructure based on IEC 61850 devices. This mapping can then be used to deploy the application to the field. After a small motivating example, this chapter will give an overview of different possibilities for reaching this goal. Finally, this chapter briefly discusses the chosen approach, while the following chapter provides additional details about the framework.

3.1 Motivating Example

The following example will be used throughout this thesis: assume that feeders of a substation are divided into several sections by using switches. If an error occurs in one of these sections, e.g., because a tree has damaged an overhead line, the goal is to shut down only the affected section and continue to supply the unaffected areas, either via the same feeder or via feeders connected by so-called tie switches. This scenario is also known as FLISR.

Figure 3.1 illustrates this situation using a simplified single-line diagram. The grid contains three feeders separated by tie switches. Each feeder is supplied by a distribution substation, which, for redundancy, typically contains more than one transformer. Furthermore, sectionalizing switches divide each feeder into sections. Assuming the incident where a tree falls on the overhead line, breaker CB1 will detect the fault current and break the circuit. Subsequently Feeder 1 is disconnected completely, even though only the first section is damaged. In order to reconnect the unaffected sections, the sectionalizing switch ROS1 has to be opened. Additionally, either tie switch ROS3 or ROS4 has to

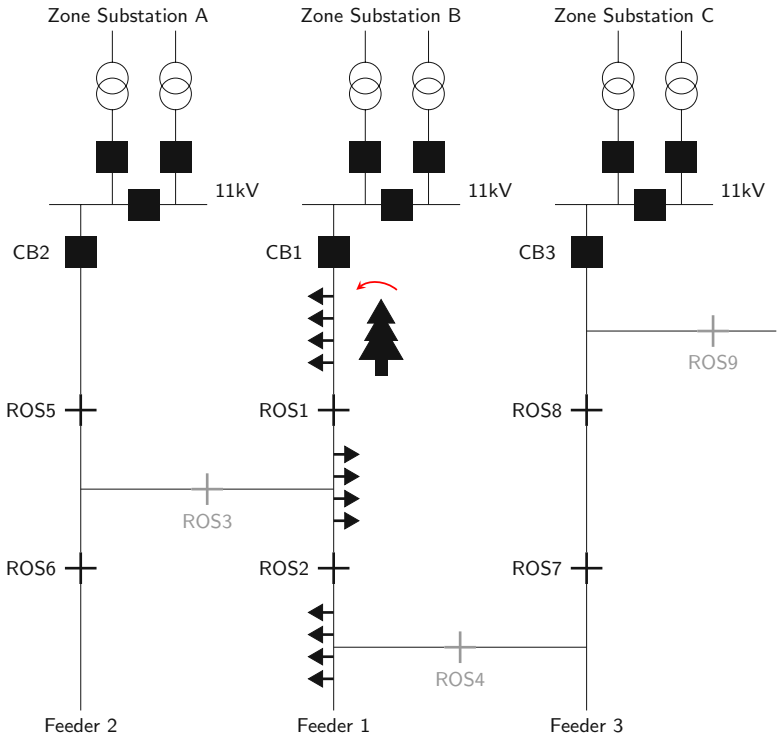


Figure 3.1: Example single line diagram of an existing grid [HVNS11]

be closed, to supply the unaffected sections. The switch to be closed is chosen by some algorithm, e.g., depending on the free capacity of the two respective feeders. [HVNS11]

3.2 General Concept and Possible Approaches

A framework that solves such problems can be built in different ways. However, there are some modules that it must contain.

Application Modeling Module: This module has to provide ways to model the desired application. In the case of this framework, this means instantiating and interconnecting IEC 61499 FBs as well as representing SG functionality in terms of FBs.

Mapping Module: In this module, the individual FBs are mapped to the real devices. In doing so, the individual device characteristics and possible constraints are to be taken into account.

Deployment Module: This module finally creates deployment files for the individual devices. These are used to deploy the mapped application in the field and configure the individual devices accordingly.

The following sections discuss possibilities and requirements to implement and interconnect the aforementioned modules.

3.2.1 Application Modeling

As already mentioned, the application engineer uses IEC 61499 for describing new SG applications. Besides common IEC 61499 FBs representing controllers or functions like optimization and database access, the application engineer can also use FBs that encapsulate IEC 61850 functionality, called IEC 61850 FBs in this thesis. Nevertheless, there are several possibilities for how an implementation of a suitable engineering tool could look. For example, the application could either be modeled non-generic automated (topology is completely known), semi-generic automated (some topology is already known), or completely generic automated (no information about the grid topology). Thereby, the less information used, the greater the implementation complexity of the framework and the application gets. This section examines a few different possible approaches.

Non-Generic Automated Solution

In the non-generic automated solution, the topology information is already available during the description of the functionality. This information can be provided, for example, by a single line diagram like Figure 3.1. As in the already existing SysGrid approach [HVNS11], FBs could automatically be created for each logical element (switch, circuit breaker, transformer). Figure 3.2 shows how an FB that represents a transformer could look like. Inputs and outputs depend on the corresponding LN. For example, 'AmpSvs' represents the sampled current. A new measurement leads to a new TRG event. Inputs like the rated current 'ARtg' are set by receiving a REQ input event and are confirmed by CNF. Initialization is done via its INIT event.

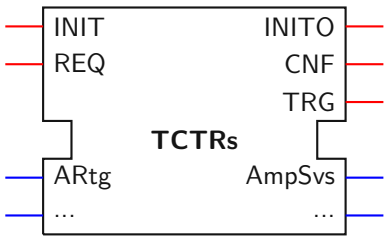


Figure 3.2: Non-generic automated description - TCTR Function Block

Besides transformers, the motivating example contains switches and circuit breakers which could be represented by XSWI-FBs or XCBR-FBs, respectively. Figure 3.3 shows how the generated FB-network for the motivating example could look like. With the help of additional FBs, the complete functionality can be described as an FB application by connecting the different FBs. Since each component is mapped to exactly one FB, the mapping is unambiguous and hence positional information is included implicitly.

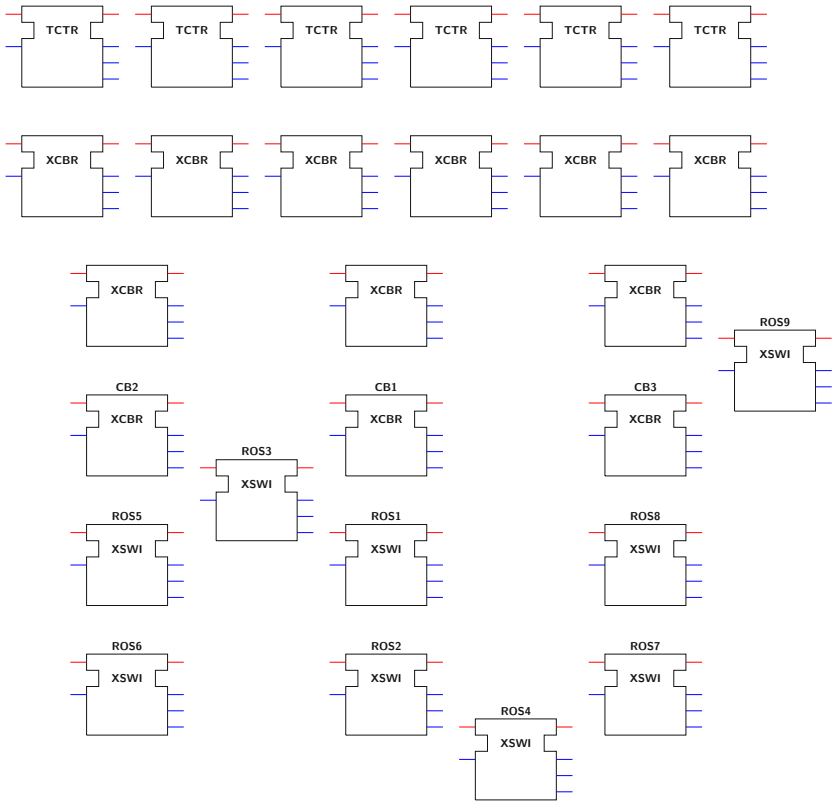


Figure 3.3: Non-generic automated description - generated FB network

Besides the unambiguousity, benefits of this approach are the low implementation complexity of the mapping module but also the simplicity for the application designer. The biggest drawback, on the other hand, is the dependency on a specific grid. This makes it hard to reuse already existing applications.

Generic Semi-Automated Solution

In the generic semi-automated solution, in a simple case, one function block again represents exactly one node. However, contrary to the previous approach, it is assumed that the application is developed independently of a specific SG. For example, when using a simplified approach, in which the location is omitted, the system could make heuristic suggestions regarding the placement of the function blocks. These placements then have to be checked and adjusted by the engineer. Considering the first part of the motivating example, if a tree falls on an overhead line, the affected feeder should be switched off as a first step. Figure 3.4 shows how this scenario could be modeled using this semi-automated approach.

TCTR again represents a transformer and PIOC an overcurrent detection function according to IEC 61850. If PIOC detects an overcurrent based on TCTR's measurement,

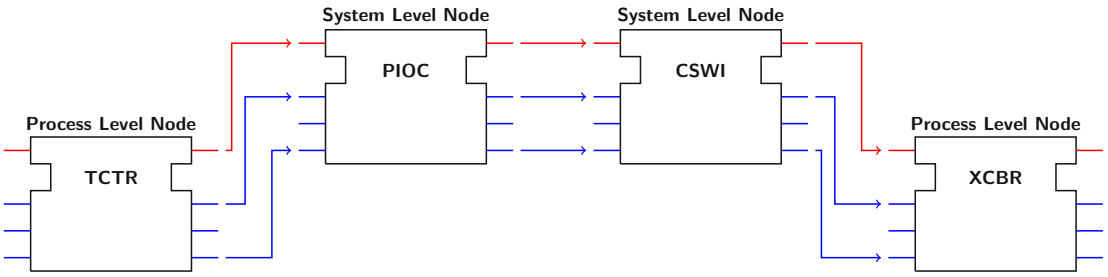


Figure 3.4: Semi automated description - close circuit breaker on over current

a switch controller (CSWI) should close a circuit breaker (XCBR). Assuming there is only Feeder 1 and only one transformer in Figure 3.1, it is easy to map the substation application. However, if there are multiple transformers, the question is how to map the application. Ideally, all transformers will be used, but this does not correspond to the assumption that one function block maps to exactly one node. Thus, multiple TCTR FBs need to be modeled with this approach.

This concept could also be extended to automatically duplicate FBs following a set of requirements, which are added as meta-information to the affected FBs. The resulting approach is similar to what has been done in project AUTEG [DPK10]. Unlike the BA area, an SG does not consist of floors and rooms. Nevertheless, SGs can also be described in a well-structured way. As IEC 61850 does, it is feasible to distinguish between different voltage levels, which in turn are divided into so-called bays. These bays are "a closely related subarea of a station with certain common functionality." [C+12b]

This breakdown could then be used to define requirements for the whole station, for all bays, for bays with specific properties, or even for individual bays. As in AUTERAS[1], a list of checkboxes could be used to identify the customer's requirements. The following list shows, how such a requirement list could look like:

- System Requirements
  - General Requirements
    - ☐ Human Machine Interface
    - ☐ ...
  - ...
- All Bays Requirements
  - Circuit Breakers
    - ☐ Switching Control
    - ☐ Connect to Protection Equipment
    - ☐ ...
  - Switching Devices
    - ☐ Switching Control
    - ☐ Connect to Protection Equipment
    - ☐ ...
  - ...

[1] https://auteras.de

The different requirements are structured by categories. Each entry is then represented by an FB application and some meta information. Figure 3.5 illustrates this concept with the requirement 'Switching Control'. In addition to the category, i.e., in which sub-item of the requirement list this requirement should appear, the type must also be specified. In case this is an aggregating requirement, i.e., the type is 'For All', the described application will only be deployed once for all required LNs. Otherwise, i.e., the type is 'For Each', it will be duplicated for each LN. In the switching control example, a separate controller will be created for each Circuit Breaker and connected accordingly. Using the format described before, this approach is extendable and adaptable.

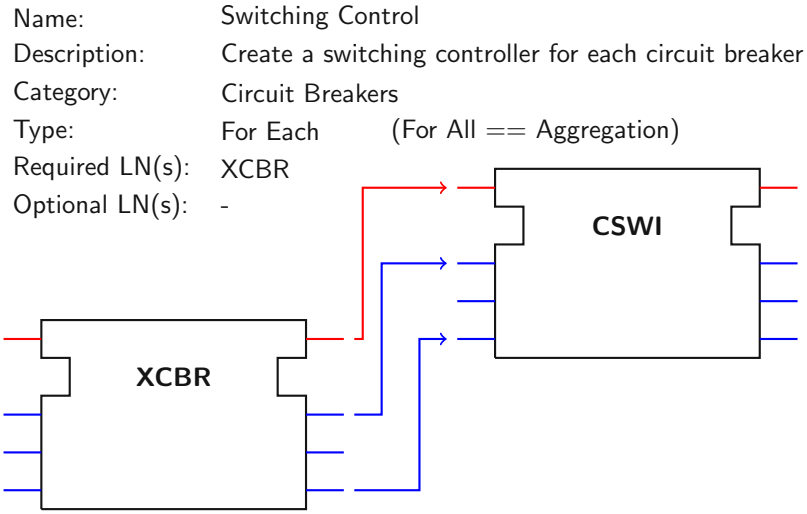


Figure 3.5: Example Requirement - Switching Control

Unlike in the non-generic automated solution, the structure of the existing grid is not required here. Basically, the application is mapped using heuristics and, hence, not unique. Therefore, the application engineer has to check the result of the mapping and adjust it if necessary. The complexity of the implementation is also higher than in the previous approach, still, one IEC 61850 FB represents exactly one component.

Generic Automated Solution

In an entirely generic solution, the application can be described independently of an existing grid, which means that it can be applied to many different grids. However, the question arises how the application represents positional relationships and dependencies using FBs, i.e., the logical position within the grid. Even for the relatively simple motivating example, it is not easy to describe which switches and which circuit breakers should operate without knowing their exact location in the grid. Therefore, additional information about sections, feeders, bays, and neighbors must be representable somehow to be able to map individual function blocks to the available equipment later in the process. For example, when looking at the FLISR scenario described before, one must

find the circuit breaker that closes the feeder on which the fault current occurs. The information can be added to FBs by defining additional inputs and outputs, as exemplified in the following.

FBs that represent process level nodes, such as TCTR (Transformer), XCBR (Circuit breaker), XSWI (Switch), are extended in a way such that events can be used to select the desired nodes. For example, by adding additional boolean inputs, it is possible to select all nodes of a specific type within a bay, all within a section, or specific devices. Accordingly, the output variables contain an array of values instead of a single value, and so do the input variables. Figure 3.6 depicts how a function block representing multiple transformers could look.

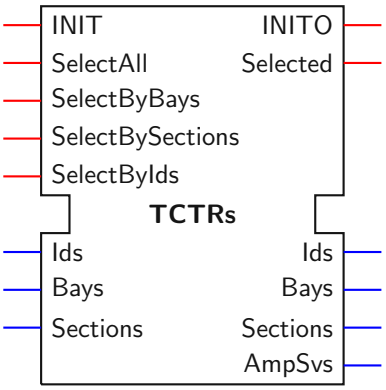


Figure 3.6: Function block that represents multiple transformers

Sectionalizing switches like ROS 1 and ROS 2 or tie switches like ROS 3 in Figure 3.1 are located in between two sections or bays, respectively. Thus, they need two separate bay/section array outputs. Figure 3.7 shows a way to represent sectionalizing and tie switches using FBs.

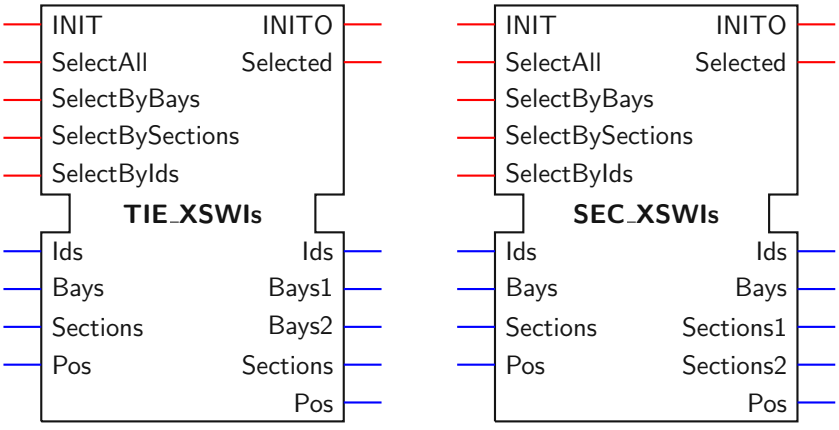


Figure 3.7: Function Block that represents tie and sectioning switches

Bay level nodes (e.g., protection or measurement function) are always connected to other bay level nodes or process level nodes (real equipment). Hence, they do not need such a selector but can work with the IDs of their trigger node. Aggregation and filter FBs can be used to modify this list of IDs.

The basic idea of this approach is, that a single FB can represent any number of functions or primary equipment of a specific type. Afterward, these are then duplicated and distributed according to the real topology. It should be noted, however, that this adds additional complexity in the device mapping module. The approach just described is entirely generic and automated as far as possible, i.e., the mapping is done automatically without the application engineer's intervention.

3.2.2 Device Mapping

Once the application has been modeled, its FBs have to be mapped to the devices. It should be noted, however, that both IEC 61499 FBs and IEC 61850 FBs can only be mapped to devices that also support them. Furthermore, any constraints resulting from the application modeling must be taken into account. These can be constraints on the FBs themselves (e.g., limited memory) or on the connections between FBs (e.g. limited communication bandwidth).

For most cases, the mapping of the FBs to the devices in compliance with the constraints is an NP-hard problem. For example, the simple limited memory constraint can be reduced to the variable-sized bin packing problem [FL86]. Basically, this problem tries to solve the question whether n objects with weights $w_1, \dots w_n$ can be packed in k buckets with sizes $b_1, \dots, b_n$. The reduction is quite simple; while FBs represent the objects n, their sizes represent the weights $w_1, \dots w_n$. Buckets b then correspond to the devices, whereas their available space corresponds to $b_1, \dots, b_n$. The Entscheidungsproblem is solved by looking at the output of the device mapper. If it finds a solution, the answer is 'YES', if not, the answer is 'NO'. Accordingly, the bigger the application gets, the harder it will be to find a best solution within a reasonable time frame. Nevertheless, there are several ways to approach them.

For example, if it is sufficient to find a valid but not necessarily the best solution, it is possible to proceed as follows: First, it is assumed that the entire solution space, hence all possible combinations, are valid. Then the solutions are checked one by one, and if it turns out to be an invalid mapping, all similar solutions are removed from the solution space. Similar in this context means all solutions that have the same property, which was the reason for their invalidity. An example would be the mapping of an FB to a device that does not support this FB. Consequently, all mappings that contain this mapping as a subset can be removed. This process is repeated until a valid solution is found. In the worst case, this approach can also lead to an exponential runtime. Nevertheless, the average case will terminate quicker.

Algorithms for iterative improvements like 'Simulated Annealing' can be used once a valid solution has been found. Also, applying heuristics, known from other NP-hard

problems, is possible.

No matter which algorithm is used, inputs cover the existing grid model, including information about the various devices, the application with its constraints, and additional application-level constraints. Furthermore, information about the installed devices is required. This information can either be already contained in the existing grid model or stored externally in some device database.

After the device mapping has found one or more valid solutions, these have to be passed on to the deployment. Again, there are several possibilities. For example, all solutions found can be stored in a database. As there are both IEC 61850 compatible and IEC 61499 compatible devices, the mapper could directly produce the individual files. For IEC 61499, a mapping could utilize system files, whereas, in IEC 61850, a mapping could be stored by using SCL files. While technology-independent storage of the mappings could also be used for frameworks in other domains, standardized files guarantee the subsequent deployment module's interchangeability.

3.2.3 Deployment

The goal of this module is to deploy the modeled application according to a selected mapping found in the mapping module. Depending on the device mapping output, this module either has to deal with standardized files from IEC 61499 and IEC 61850 or with mappings stored in some database.

As already mentioned, there are two types of FBs, which also need to be supported by the deployment module. IEC 61850 FBs encapsulate IEC 61850 functionality. The IEC 61850 standard already provides a standardized way to configure the corresponding devices by using CID files. An advantage of this approach is that the existing configuration remains untouched. In addition it would also be possible to configure these IEC 61850 devices over the network using GOOSE.

Also IEC 61499 provides a standardized way to configure devices. All IEC 61499 devices run a runtime environment. This is a program executing IEC 61499 applications and allows the configuration and programming of the device. Management commands are used to instantiate new FBs and connect them accordingly. Furthermore, there are management commands for starting and stopping the applications, as well as watching runtime values. An example of such a runtime environment is 4DiacForte[2]. It also offers the possibility to pass the management commands to it using text files. A disadvantage of this approach is that the standard does not specify the execution of the code encapsulated in the FB. For example, with the 4DiacForte runtime environment, the respective FB must already be compiled with the runtime environment in order to be able to instantiate it afterwards.

[2] https://www.eclipse.org/4diac/

3.2.4 Additional Requirements and Considerations

In addition to three main modules – application modeling, device mapping, and deployment – a complete framework for modeling applications in the SG domain needs to cover numerous additional requirements and cosiderations. These are briefly discussion in the following.

Independently of the specific application modeling solution chosen, all three approaches need some source for the IEC 61850 FBs. For example, it would be possible to generate them in advance and make them available as an IEC 61499 library. It would also be possible to make them available directly in a modeling tool via a database or an ontology. While the first option offers the advantage of using any existing IEC 61499 modeling tool, the second option offers greater flexibility. So, it would be conceivable to generate FBs automatically depending on the respective requirements.

Possible constraints are another point that is important when modeling the application. A distinction can be made between FB-level, connection-level, and application-level constraints. Examples for FB- and connection-level constraints are reliability and transmission speed, respectively. Some constraints can already be specified during application development, while others like deploying an application on as few devices as possible can only be specified at the device-mapping stage. Besides storing the possible constraint types, also the instantiated constraint types have to be stored. These instantiated constraint types could be stored either directly in the IEC 61499 files or externally, depending on whether standard-compliant files can still be obtained if the additional information is stored. The available constraint types can be stored as a fixed list in a simple case, or in a more complex scenario using an ontology.

An application repository could then be used to store multiple different modeled applications. This easily allows the deployment of applications to various SGs. The repository could either be as simple as a folder or some database that allows location-independent access.

In the next step, a modeled application is to be mapped to one existing infrastructure. Since this infrastructure is already limited to IEC 61850 devices, it makes no sense to look for alternatives to a representation based on IEC 61850. Therefore SCD files from IEC 61850 are used for the representation of the infrastructure. However, theoretically, also any other alternative representation could be adapted for this purpose. The only requirement is to be able to represent the current grid, including the already existing infrastructure.

As mentioned, information about the installed devices is required in the device mapper. This information can either be stored directly in the model of an existing grid or using an explicit database. For future extension this information is stored externally. This allows, for example, the selection of suitable devices in order to propose new solutions.

Finally, the deployment module is responsible for the deployment of the selected application to devices in the grid. In order to test and validate the created applications, it

should also be possible to deploy the application to pseudo-IEC 61850 devices. These for example could be docker containers, that use libraries such as libIEC61850[3] and additionally run an IEC 61499 runtime environment like 4DiacForte.

3.3 Framework Definition

The three main modules and the additional requirements and considerations can now be combined into a complete framework supporting the design, mapping, and deployment of SG applications. The framework is illustrated in Figure 3.8. On the left, the application designer functionally describes a desired new SG application. This functional description will be done using FBs from IEC 61499, and the Semi-Automated Solution described before in Section 3.2.1. The first approach, the non-generic solution, will not be pursued because a goal is decoupling the functional description from the existing model, i.e., the possibility to apply a described use-case to different grids. Even though the third approach is completely generic, the implementation complexity as well as the design complexity is high and will thus be considered as future work. Hence, the semi-generic approach is used as proof of concept, which can then be easily extended to the described requirement list.

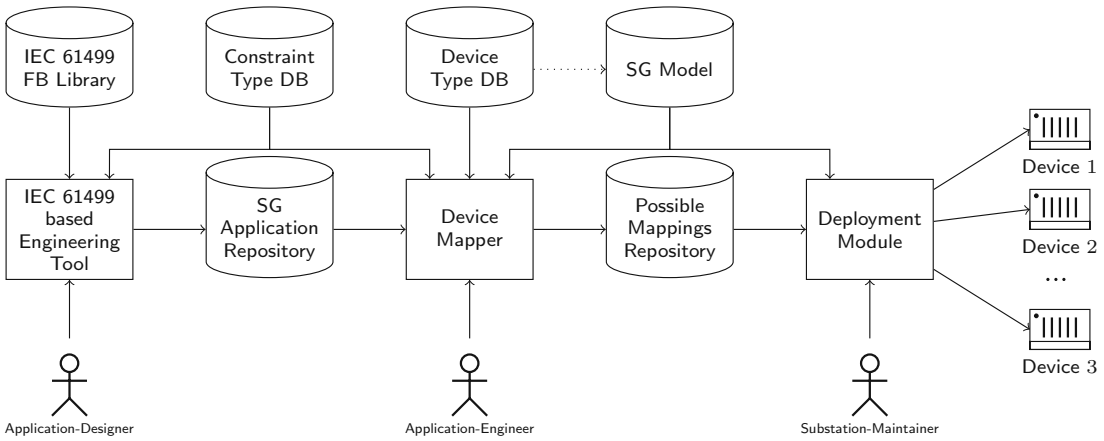


Figure 3.8: Framework for the design, mapping, and deployment of SG applications

Additionally to IEC 61499 FBs from the standard library, LNs from IEC 61850 are represented by FBs from IEC 61499, similar to [HVNS11]. Those IEC 61850 FBs are used to represent functions and primary equipment of the SG area. Furthermore, the application designer can define constraints at this stage, e.g., how reliably a function should be executed or within which time frame. The finished applications will then be stored in the SG application repository.

The application engineer can map an application from the repository to an existing smart grid in the next step. Here it is possible to specify the application and system-level

[3] https://libiec61850.com/

constraints. The Device Type DB provides meta-information as well as the capabilities of all available devices in the SG. By separating the SG model from the device type DB, it is also possible to extend the mapping algorithm such that it proposes additional devices if the application cannot be mapped to the existing infrastructure. However, this is currently not implemented but left open for future work.

The device mapping module then tries to map the selected functional description, i.e., the application, to the corresponding, already existing device infrastructure while ensuring the best possible compliance with all given constraints. Valid solutions, i.e., possible mappings, will then be stored in a database.

A substation maintainer can choose one of the provided mappings and can further adjust them if needed. Afterward, the complete description of the substation, including deployment files for the individual devices, will be generated and deployed in the field.

CHAPTER 4

Framework Components

The general structure of the framework was presented in the previous chapter. It was determined that the framework has to consist of at least three modules:

Application Modeling Module provides ways to model FB applications

Device Mapping Module maps the individual FBs to the real devices

Deployment Module based on the found mappings creates the deployment files for the individual devices

This chapter describes the details of the individual components of this frameworks, as well as their interfaces. Thereby, the affected part of the overall framework is also shown in each case.

4.1 IEC 61499 FB Library

As already mentioned in the last chapter, this thesis will use the semi-automated approach for describing SG applications. Figure 3.4 already provided an abstract example consisting of only very few FBs. There are two types of FBs:

IEC 61850 FBs encapsulate SG functionality

IEC 61499 FBs represent controllers or functions like optimization and database access

This section shows how the IEC 61850 FBs can be represented and automatically generated. Furthermore, it gives an overview of the IEC 61499 XML representation of FBs. Figure 4.1 shows the location and connections of this component within this framework.

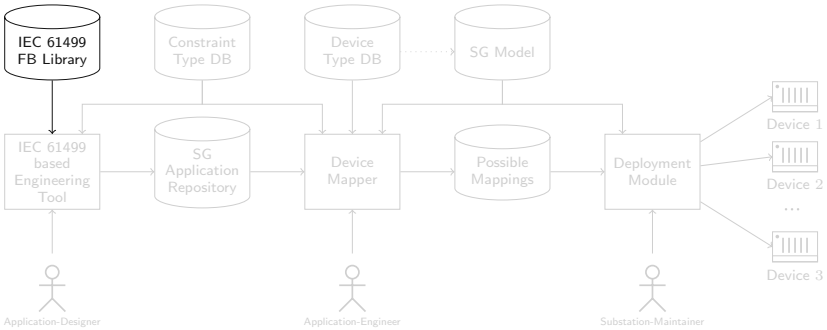


Figure 4.1: Location of the IEC 61499 FB Library within the Framework

4.1.1 IEC 61850 FBs

The idea is to model IEC 61850 FBs as SIFBs, which encapsulates the services provided by the LNs. As events, INIT and INITO are used for initialization, and REQ and CNF are used for a request and confirmation, respectively, as suggested in the IEC 61499 standard. Moreover, the output event TRG is introduced, which triggers as soon as a value changes. The proof-of-concept will currently be limited to reading and writing values of the LNs. CDCs of the corresponding LNs, which are used as inputs and outputs, are represented as structured data types. While the structured data types can be automatically generated based on the CDC tables from IEC 61850, it is not that easy for the IEC 61850 FBs. IEC 61850 only provides a standardized data model. The implementation of LNs is vendor dependent. This means for example that algorithms from different vendors may need different inputs. Thus, the required inputs of the FBs, i.e., the subscriptions of the LNs, are not standardized. So manual adaptions may be required later on.

Figure 4.2 illustrates an FB that represents a PIOC LN. Depending on the Functional Constraints (FCs), Data Objects (DOs) of LNs are used as input or output variables. As mentioned, the IEC 61850 specification defines mandatory and optional attributes for each LN. A viable approach is to include both types of attributes in the FB representation of the LN. Based on the assignments of the finished application, required attributes can be identified to find suitable devices. Additional, commonly used inputs depending on the respective LN have to be added manually. In the case of PIOC, the measured current from TCTR or MMXU might be required.

4.1.2 IEC 61499 FBs

It should also be possible to connect the IEC 61850 FBs with regular FBs from IEC 61499. Furthermore, the implementation of system-level nodes has to be possible as well. For this purpose, a small library consisting of some BFBs and the mentioned IEC 61850 FBs was created for this framework. This library provides FBs that facilitates the (de-)composing of the structured data types as well as the automatically generated data types, but also the generated IEC 61850 FBs. Figure 4.3 shows two example FBs that compose and

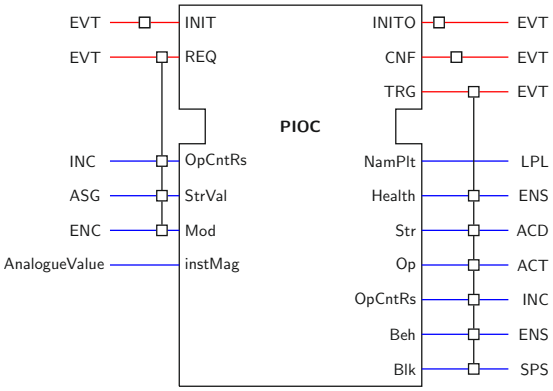


Figure 4.2: Function Block representing a PIOC Logical Node

decompose the structured types Analogue Setting (ASG) and DPC, respectively. Besides request and confirmation events, they include all DOs of the corresponding CDC as input or output. In case of composition FBs, the structured data type will be produced, while in case of decomposition FBs the structured data type will be split up. In the example depicted, COMPOSE_ASG produces an analogue set point of 12 amperes.

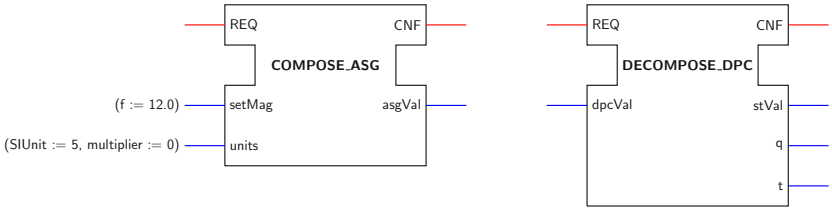


Figure 4.3: Function Blocks for (de-)composing the Structured Data Types

4.1.3 Representation of FB library

Since IEC 61499 offers the XML representation of all its elements, an IEC 61499 FB library is represented as a collection of XML files in this framework. Before showing how the automatic generation of the structured data types and the FBs takes place, an explanation is given of how IEC 61499 represents its elements by XML. Figure 4.4 shows the structure of data type files. Besides some meta-information like 'CompilerInfo' and 'VersionInfo', it can be seen that the standard defines five different types:

Directly Derived Type represents an alias for one of the already existing types

Enumerated Type represents a simple enumeration

Subrange Type represents a numeric interval

Array Type represents a collection of same type values

Structured Type represents the composition of different data types

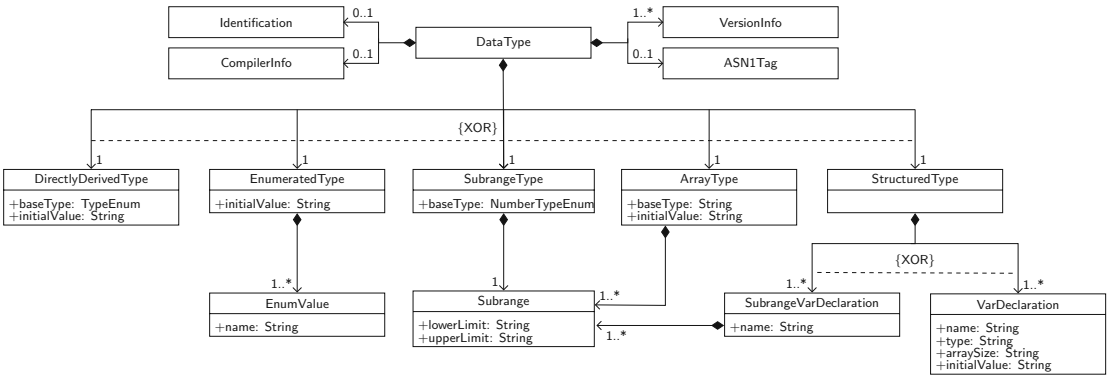


Figure 4.4: IEC 61499 Data Type Structure [C+10]

Figure 4.5 shows how FB types can be represented using XML. In addition to meta information, each FB type contains an interface list, i.e., a list of all data and event inputs and outputs, as well as all plugs and sockets. Especially in SIFBs, it is possible to describe the service sequences of the FB types. BFBs and CFBs can be mapped using BasicFB and FBNetwork, respectively, whereby an FB cannot be mapped to both.

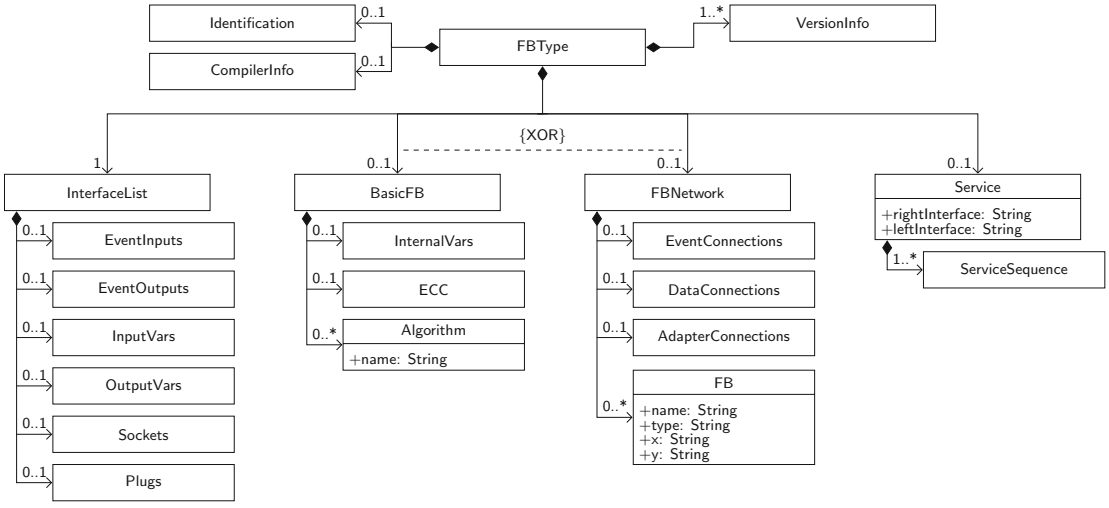


Figure 4.5: IEC 61499 Function Block Structure [C+10]

4.1.4 Automatic FB Generation

Based on the XML representation, it is straight forward to imagine the automatic generation process. First, all 'simple' data types from IEC 61850 are mapped. For example, strings and boolean values can be mapped immediately as Directly Derived

Type, whose base type is a string or boolean, respectively. Quality and Timestamp types are defined according to IEC 61850 7-1/7-2 as Packed List and Composite Type. Therefore, it is reasonable to represent them as Structured Data Types in IEC 61499. Table 4.1 gives an overview of how the IEC 61850 types are represented in IEC 61499. Subsequently, all CDCs are then composed as Structured Data Types from the created 'simple' data types. Appendix 7.2.1 illustrates the procedure for the SPS CDC.

IEC 61850 Type	**IEC 61499 Data Type**	**Remark**
BOOLEAN	DirectlyDerivedType(BOOL)	
INT8	DirectlyDerivedType(SINT)	
INT16	DirectlyDerivedType(INT)	
INT32	DirectlyDerivedType(DINT)	
INT64	DirectlyDerivedType(LINT)	
FLOAT32	DirectlyDerivedType(REAL)	
INT8U	DirectlyDerivedType(USINT)	
INT16U	DirectlyDerivedType(UINT)	
INT24U	SubrangeType(UDINT)	LowerLimit: 0, UpperLimit: $2^{24}-1$
INT32U	DirectlyDerivedType(UDINT)	
ENUMERATED	EnumeratedType	
CODED ENUM	EnumeratedType	
OCTED STRING	DirectlyDerivedType(STRING)	Including length
VISIBLE STRING	DirectlyDerivedType(STRING)	Including length
UNICODE STRING	DirectlyDerivedType(STRING)	Including length
ARRAY	ArrayType	
PACKED LIST	StructuredType	
Other composed types	StructuredType	

Table 4.1: IEC 61850/61499 Data Type Mapping

The automatic generation of the FBs is done as follows: Based on the LN tables from IEC 61850, CDCs with the FC Settings become input variables, CDCs with the FCs Status Information, or Measured and Metered Values become output variables, and controls become both input and output variables. The events described above are created and linked accordingly and described by using ServiceSequences. Again, an example using the PIOC LN can be found in Appendix 7.2.2. Afterward, commonly used inputs have to be added manually in a simple case or could be automatically generated based on heuristics or some kind of knowledge graph in a future version.

4.2 Constraint Type DB

Besides the IEC 61850 FBs, the engineering tool also allows to instantiate constraints. This section specifies which constraints are available and what they look like. Figure 4.6 shows the location and connections of this component within this framework.

A distinction can be made between different types of constraints:

FB-Level Constraints describe requirements for algorithms included in the FBs, repre-

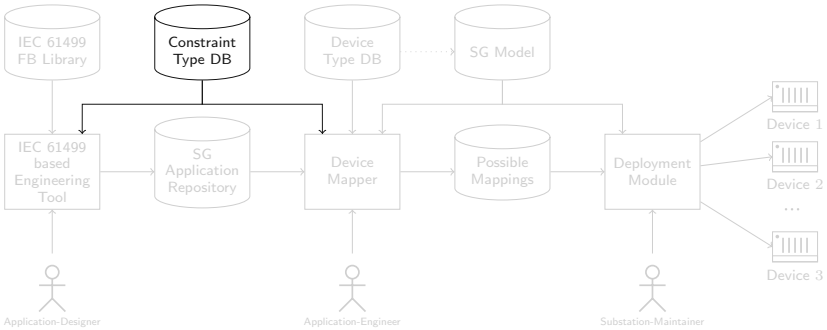


Figure 4.6: Location of the Constraint Type DB within the Framework

sented LNs, inputs, and outputs. An example would be an LN FB whose algorithm must return certain DOs or Data Attributes (DAs).

Connection-Level Constraints include all the requirements related to the connections such as the required link speed.

Device-Level Constraints represent requirements on the FBs, as well as the devices. An example is the check of the memory usage.

Application-Level Constraints describe holistic requirements like using as few devices as possible.

While the first three types are typically already employed during application modeling, application-level constraints are usually specified within device mapping. Each of these constraints can then either be asserting or accumulating. For example, the check whether a device supports IEC 61499 FBs is an asserting device-level constraint, while the check whether a device has sufficient memory is an accumulating device-level constraint. Here, the memory consumption of the individual FBs mapped to the device has to be summed up first, to then compare the sum with the available memory. Furthermore, the checks on the different levels have to be specified, i.e., depending on the constraint type, either a statement that returns a boolean or a numeric value.

These checks then have to be parsed and executed by the device mapper. Besides simple arithmetic operations, comparison operations are also permitted. In addition, the keywords IED and FB allow access to the device and the FB. With a dot or brackets the individual attributes of the instances can then be accessed. The ref function is used to access references like the FB type of and FB instance. Table 4.2 shows how the two mentioned constraints could be represented. Besides the name, the level, and the type of a constraint, it also states how the checks could look like. Here, for example, `IED.Private("Memory")` accesses the memory field of the IED, while `FB.Size` accesses the size of an FB. Checking this constraint then requires the device mapper to call the FB check for every FB mapped to the device, sum it up and compare the sum to

the available memory checked by the device check. The available fields will be discussed in more detail.

Name	Free Space Constraint	IEC 61499 Compliance Constraint
Type	Accumulating	Asserting
Level	Device-Level	Device-Level
Device Check	IED.Private("Memory")	IED.Private("Programmable") == True
FB Check	FB.Size	ref(FB.Type).Identification.Standard == "IEC 61499"

Table 4.2: Constraint DB Example Constraints

While a real database would probably be preferable for productive use, this framework relies on a description in XML for simplicity. Figure 4.7 shows the corresponding structure used in this framework, while the XML Schema Definition (XSD) can be found in Appendix 7.1.1. A proper XML then contains a list of the different constraint types and, depending on the type, the different checks and their names. This format easily allows the definitions of the constraints shown in Table 4.2.

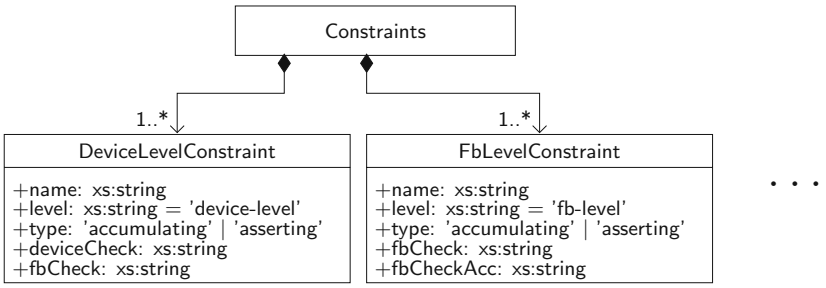


Figure 4.7: Constraints XML Structure

4.3 IEC 61499 based Engineering Tool

The task of the engineering tool is to create the new SG applications. It is possible to instantiate IEC 61850 FBs as well as IEC 61499 FBs and to connect them. In addition, constraints can be specified. The result of this module will then be stored in the application repository. Figure 4.8 shows the location and connections of this component within this framework.

As illustrated, IEC 61499 provides a uniform XML-based description for datatypes and FB types. The standard also offers Document Type Definitions (DTDs) for the description of entire systems. These DTDs form the basis of the engineering tool. Figure 4.9 shows the structure of a system. A system can contain instances of devices, which in turn can be composed of multiple resources. These resources then contain the distributed FBs including the generated communication FBs. Depending on the device, resources can

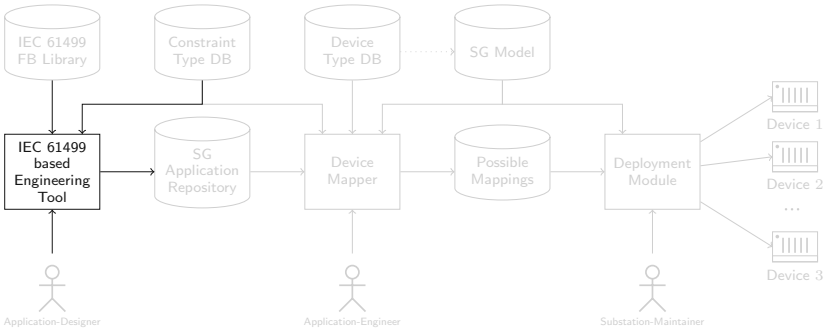


Figure 4.8: Location of the IEC 61499 based Engineering Tool within the Framework

represent processes or even virtual machines. The mapping section then maps the FBs of the application to the corresponding FBs in the resources and, thus, describes the distribution. Furthermore, a system contains segments, i.e, network segments of a certain type such as Ethernet, and links, i.e., which resources are connected to which segment.

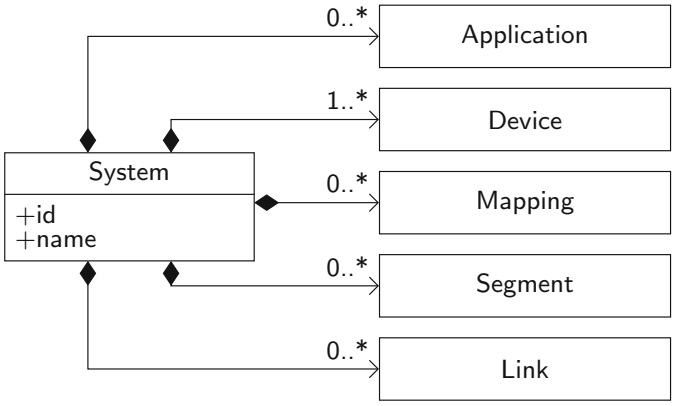


Figure 4.9: IEC 61499 - System structure [C+10]

For the engineering tool, the application section, is the relevant part. Figure 4.10 shows how an application is structured. It consists of a network that contains FBs, subapplications, and connections. FBs and subapplications are instances of types specified as XML-files in the IEC 61499 FB Library. The comment field, which is omitted for clarity in the figure, of the elements Application, FB/SubApp, and Connection is used to specify the application-level, FB-level, and connection-level constraints to be used. The complete DTDs and their detailed descriptions can be found in Part 2 of the IEC 61499 standard.

As Figure 4.9 shows, a system must contain at least one device. But in this framework the idea is to describe the application independently of an existing grid. This means no device is known at this stage. One solution to this problem would be to use only the application section, which is not a file on its own according to IEC 61499, i.e., there is

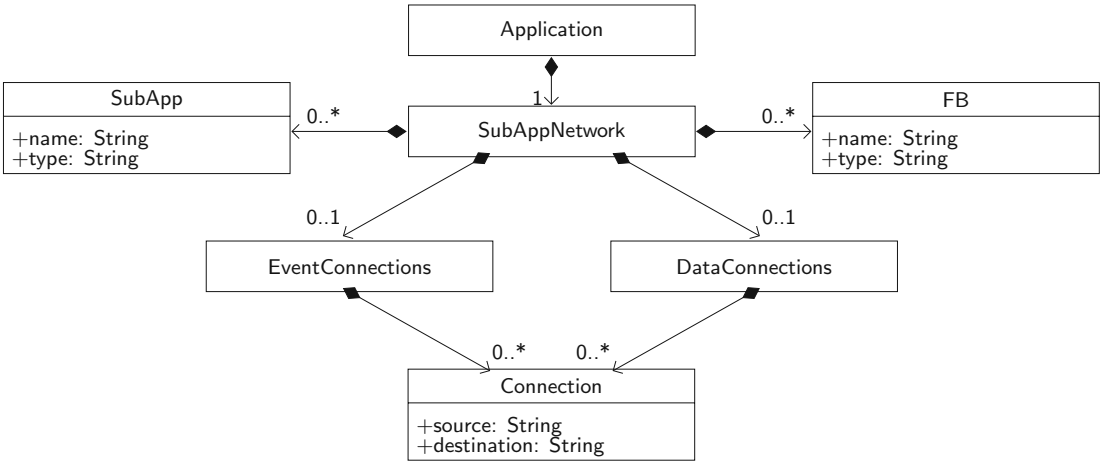


Figure 4.10: IEC 61499 - Application structure [C+10]

no DTD for applications. Another possibility is to create an empty template device that has the type 'TEMPLATE'. In this thesis, the second approach is chosen in order to keep standard conformance. This allows the use of already existing tools like 4Diac[1] to create the system files.

In the engineering tool, it is then possible to connect the various FB instances according to IEC 61499 to create the desired applications. These can then be stored in the application repository as system files.

4.4 SG Application Repository

The SG Application Repository is the interface between the engineering tool and the device mapper. Figure 4.11 shows the location and connections of this component within this framework. It stores a collection of IEC 61499 system XML files. These contain one

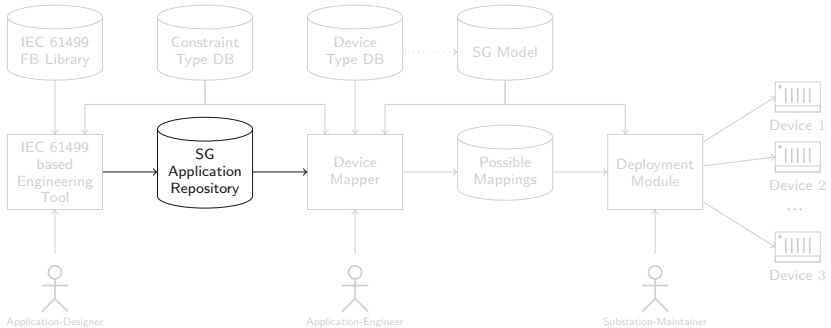


Figure 4.11: Location of the SG Application Repository within the Framework

[1] https://www.eclipse.org/4diac/

or more applications, a TEMPLATE device, and constraint information created by the engineering tool and read by the device-mapper module. In this framework, for simplicity, a simple folder is used to store these files. In productive use, this could become a publicly accessible database. This would allow the sharing and use of already modeled use-cases within the community.

4.5 Device Type DB

This section describes the structure of the Device Type DB, where the specifics of the various devices of different vendors are stored. Like IEC 61499, IEC 61850 also offers a standardized XML-based description language, called SCL. In contrast to IEC 61499, however, the description is provided in the more powerful XSD format. Figure 4.12 shows the location and connections of this component within this framework.

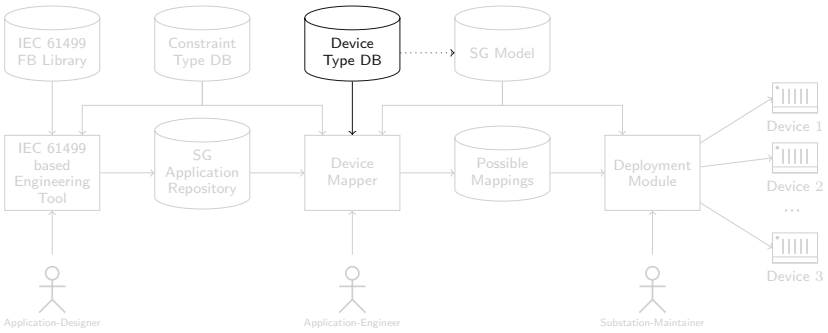


Figure 4.12: Location of the Device Type DB within the Framework

An SCL file consists of several sections, as can be seen in Figure 4.13. While SCL files allow the description of a complete substation, including its devices and infrastructure, IEC 61850 also provides files that focus on specific aspects, i.e., they represent a subset of SCL. Depending on the respective file type (see Section 2.3.3), some sections are mandatory, while others are optional.

Table 4.3 illustrates this for a few selected file types. In a System Specification Description (SSD) file, which describes the substation's single line diagram, only the header, and the substation section are required. On the other hand, an IED Capabilities Description (ICD) file, which describes an IED's capabilities, usually needs the description of its communication capabilities and its supported LNs, but not the description of the substation.

In this framework, the Device Type DB is represented by a collection of ICD files. Besides the header, ICD files have to contain the IED section, depicted in Figure 4.14. A device

[2] https://cimug.ucaiug.org/Harmonization Documents/EPRI Harmonization Project Notes and Minutes/IEC TC57 Substation Configuration Language Summary.pdf

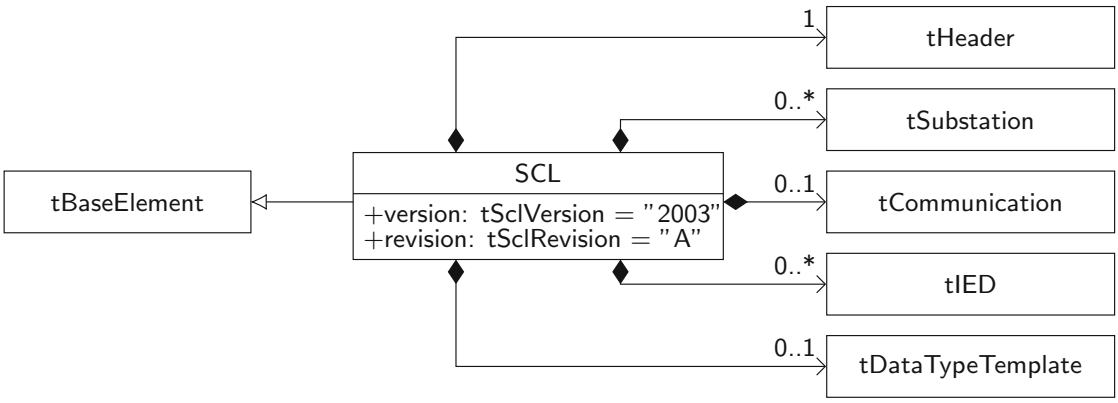


Figure 4.13: IEC 61850 - SCL structure [C+12b]

Section	File Type			
	SSD	**ICD**	**SCD**	**CID**
	Substation One-Line and Functions	IED Template	Complete Substation	Particular IED Configuration
Header	Yes	Yes	Yes	Yes
Substation	Yes	Optional	Yes	Optional
Communications	Optional	One Instance	Yes	One Instance
IED	Optional	Yes, values optional	Multiple	Yes, including values
Data Type Template	As needed	As needed	Multiple	As needed

Table 4.3: IEC 61850 - Substation Configuration Files [C+12b][2]

offers services like reading and writing, for example, log files or reading single values from the dataset. Furthermore, devices offer one or more access points, which are then used to assign the device to a subnetwork within the communication section. An IED containing a router function connects different subnetworks utilizing all its access points and acts as a limiting border for real-time message types. Also, an IED contains one or more servers that can be used to access the LNs and LDs. LNs of access points to which no server has been assigned to cannot be accessed. The LNs, in turn, contain a dataset as well as DOs and DAs. Additionally, values can already be assigned to them. The exact format of the DOs and DAs is described in the Data Type Template section. This section is necessary since the standard does not require manufacturers to implement optional attributes and objects. IEDs not yet configured are identified by their name, 'TEMPLATE'. Such template IEDs are used in this approach to build the device type database.

Besides the text tag in the header in which any content can be placed even from other namespaces (see Figure 4.15), IEC 61850 also offers private elements, which can be used throughout the XML document for small extensions. Even when imported or exported

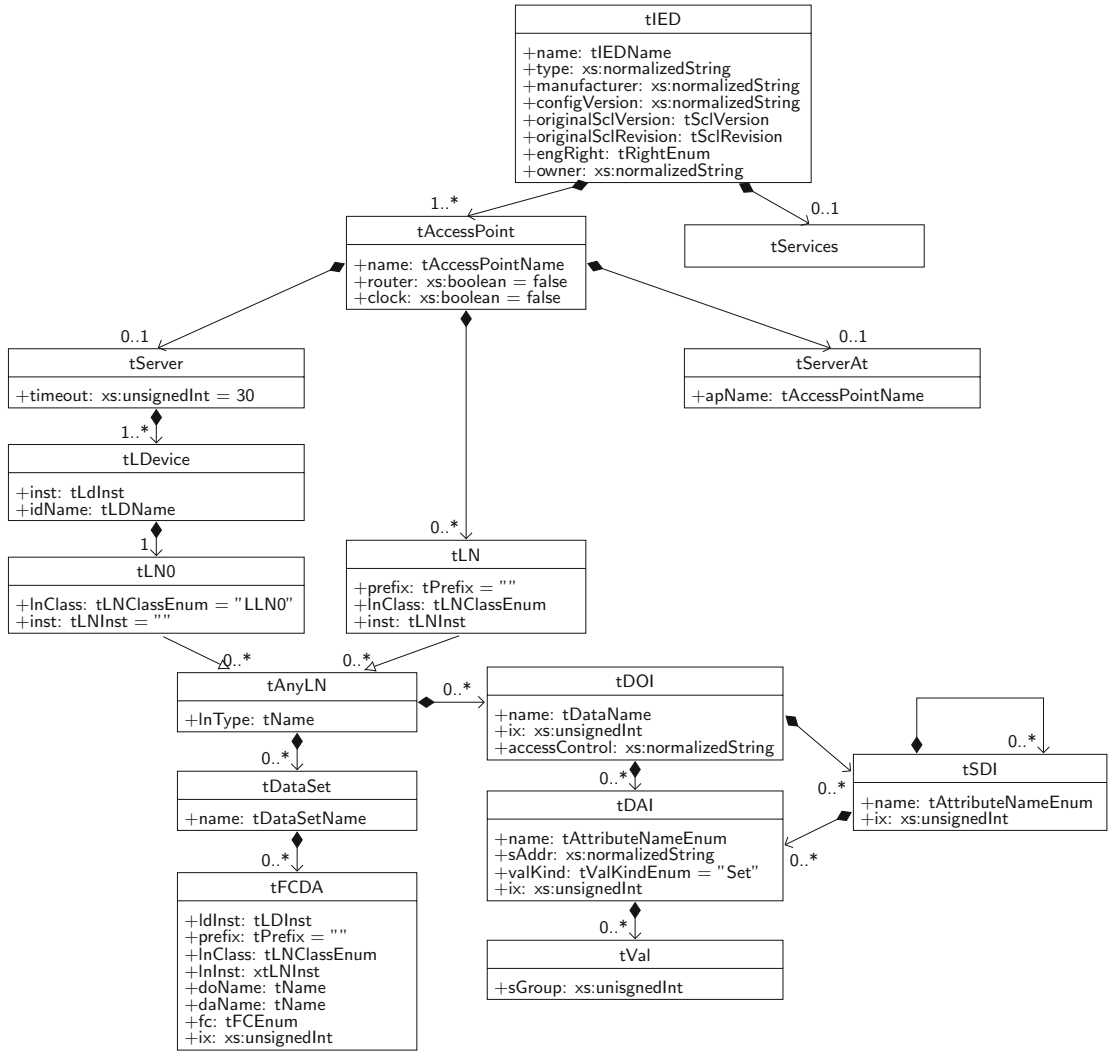


Figure 4.14: IEC 61850 - IED section structure [C+12b]

by third-party tools, it is guaranteed by the standard that their content is preserved.

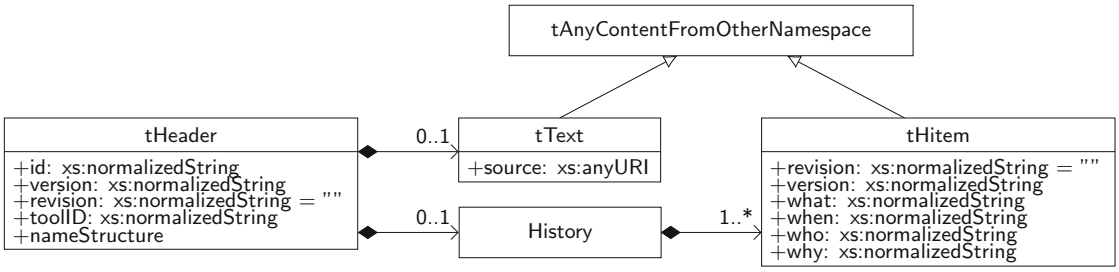


Figure 4.15: IEC 61850 - Header section structure [C+12b]

These private elements are used to indicate whether this IED only supports the corresponding LNs or the device is also freely programmable, i.e., in the context of the new approach, whether it is possible to distribute IEC 61499 FBs to this device. This private type is called 'Programmable' and is illustrated in Listing 4.1. Devices that do not have this element are assumed not to be freely programmable. Besides, meta information, such as the available memory, is placed here, which is needed for constraint optimization.

```
<IED ...>
   <Private type="Programmable">
      true
   </Private>
   <Private type="Memory">
      1MB
   </Private>
   ...
</IED>
```

Listing 4.1: Private type used in IED section

The structure of the Data Type Template section is depicted in Figure 4.16. This section,

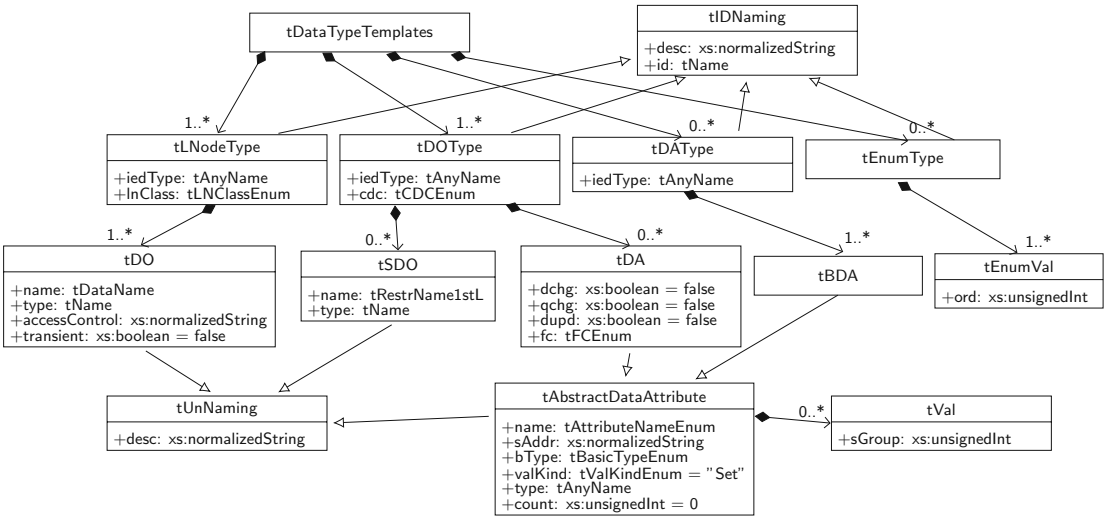


Figure 4.16: IEC 61850 - DataTypeTemplates section structure [C+12b]

which the IED section uses, describes the types that can be instantiated, such as LNs, DOs, and DAs. This section is necessary because the standard allows customized and user-defined LNs and because some objects and attributes are optional. For example, device A could use the MMXU LN to measure only the total active power, while device B could also measure the reactive power and frequency. This scenario would lead to creating two instances of type MMXU with corresponding entries of the DOs.

Currently, the Device Type DB is only used indirectly via the model of the existing grid. For a first step, in this thesis, applications are only distributed among existing devices.

Nevertheless, in order to be able to make automated suggestions for required devices in future work, this component is already included.

4.6 SG Model

The SG Model not only describes the single line diagram of the substation but the complete existing substation, including its network infrastructure and its existing devices. Therefore, this database is represented by a collection of SCD files in this framework. Here, all sections are required. Thereby, the IED and Data Type Template section are filled by instantiating devices from the Device Type DB. Figure 4.17 shows the location and connections of this component within this framework.

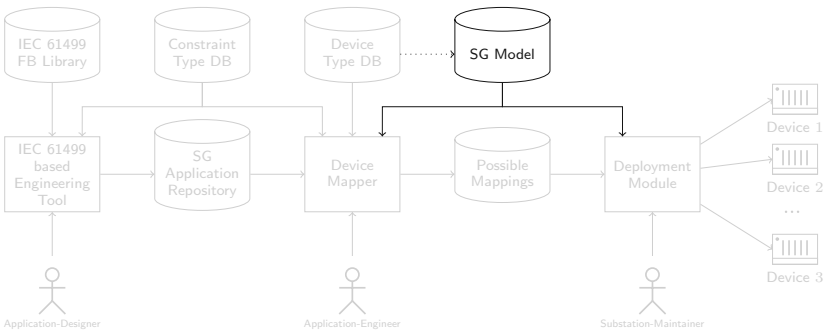


Figure 4.17: Location of the Model of existing Grid within the Framework

The substation section (see Figure 4.18 for a small excerpt) describes both the existing equipment and the existing functions, such as protection and measurement. For this purpose, the substation is divided into different voltage levels, such as high voltage and medium voltage. These, in turn, are divided into bays. According to the standard, bays are "a closely related subarea of a station with a certain common functionality" [C+12b]. Bays can contain equipment, sub-equipment, functions, and connectivity nodes used for connecting nodes within a bay or connecting several bays within the same voltage level. Conducting equipment can only reside within bays. Here, the type-code determines what type of primary equipment is involved. Those different codes are defined in part 6 of the IEC 61850 standard. For example, 'CBR' represents a circuit breaker with two terminals, i.e., connection points, while 'BAT' represents a battery and has one connection point. Functions and LNs can be located at any level of the substation.

The communication section is structured quite simple. Buses are defined as subnetworks that can contain any number of access points. These are described within the individual IEDs. Besides, the actual address on the respective bus, i.e., in the subnetwork, is also described in this section. Figure 4.19 shows an overview of the communication section. For the new approach, this section is only compulsive in so far as it allows to determine which devices can communicate with each other. This information needs to be considered when mapping FBs to devices, since it may limit the distribution of the FBs.

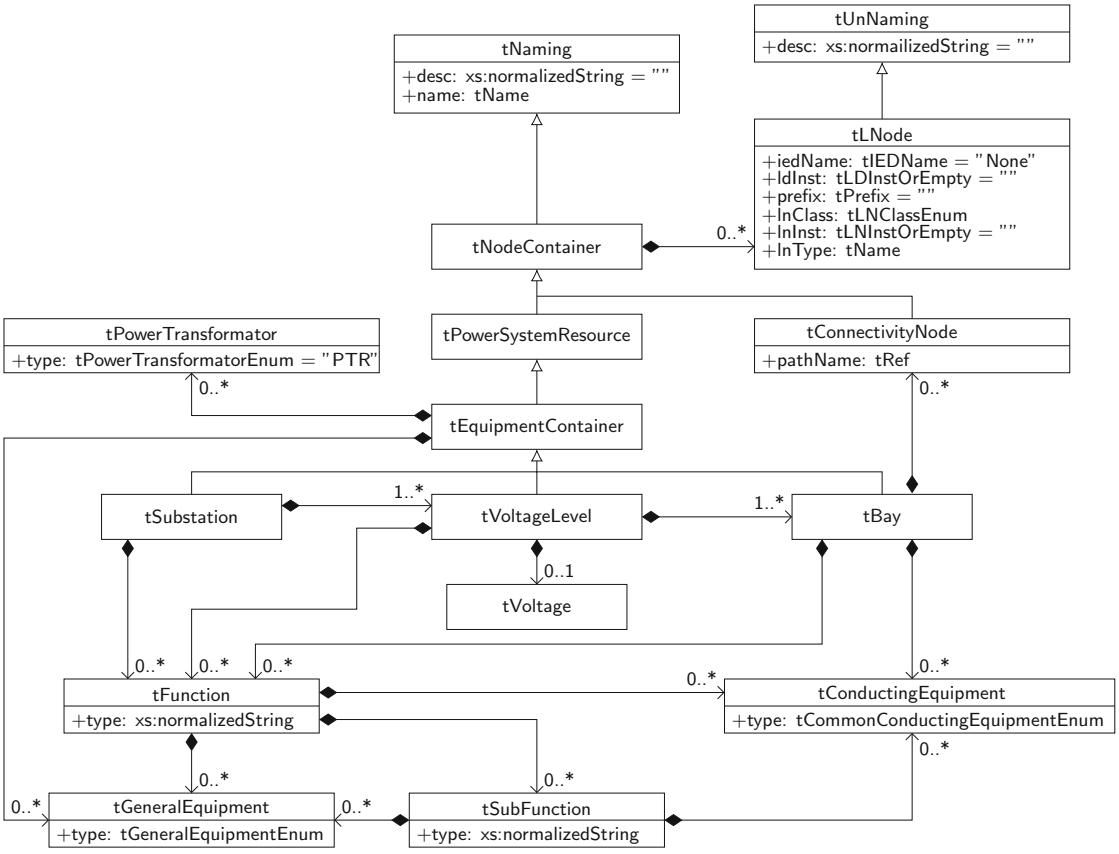


Figure 4.18: IEC 61850 - Substation section structure [C+12b]

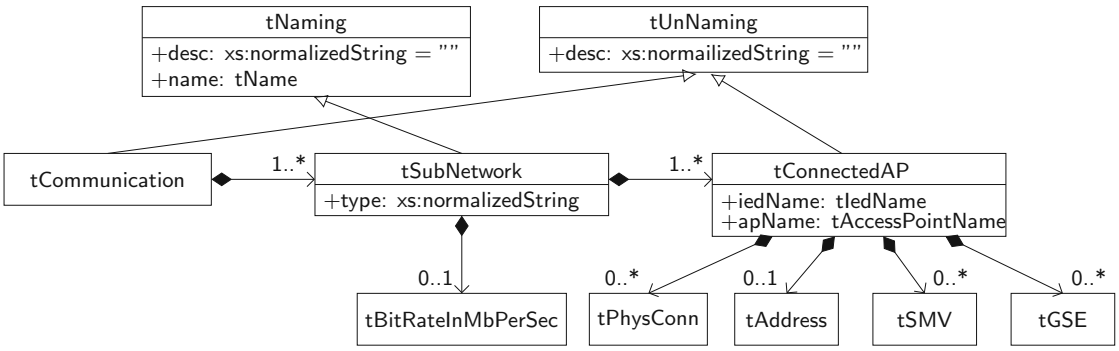


Figure 4.19: IEC 61850 - Communication section structure [C+12b]

4.7 Device Mapper

After the application engineer has chosen an application from the SG Application Repository and an existing grid from the SG Model to map the individual FBs to the entire infrastructure in compliance with the constraints. Figure 4.20 shows the location

and connections of this component within this framework.

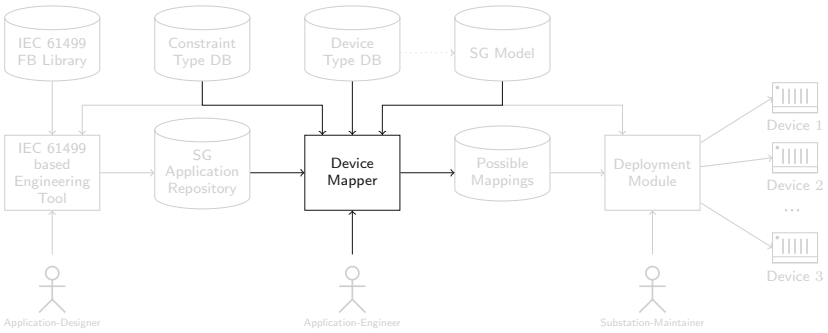


Figure 4.20: Location of the Device Mapper within the Framework

Since both IEC 61499 and IEC 61850 provide their data models via DTDs and XSDs, respectively, it is feasible to create an object model from these automatically. In Java, this could be a collection of classes provided by XJC, a tool of the Java Architecture for XML Binding (JAXB)[3]. Of course, other programming languages have similar tools. These classes correspond to the UML class diagrams shown in the previous sections. Before the mapping process can begin, the IEC 61499 library, i.e., all available FBs and data types, as well as the IEC 61850 SCL XML file and the IEC 61499 System XML file, must be read. While the SCL file represents the Model of existing SG, including the manufacturers' data, the system file represents the created FB application. The framework overview (see Figure 3.8) shows this connection.

Algorithm 1 describes the overall mapping procedure. In the beginning, the solution space will be generated for each bay with the Cartesian Power's help. However, it should be noted, that the explicit generation of the solution space only makes sense for small applications and is done here for clarity. Let $X = \{IED_1, IED_2, \dots IED_n\}$ be the set of possible devices (IEDs); then the m-ary Cartesian Power

$$X^m = \underbrace{X \times X \times \cdots \times X}_{m} = \{x_1, \dots, x_m \mid x_i \in X \textit{ for every } i \in \{1, \dots, m\}\}$$

provides all possibilities to arrange m IEDs in any order, whereby an IED can be used several times. If m corresponds to the number of FBs to be distributed, the resulting tuples' indexes can be interpreted as the i-th FB, while the i-th value can be interpreted as the IED on which this FB will be mapped. Line 9 and 10 of the algorithm do precisely this. First, assuming there are three IEDs and three FBs, the Cartesian Power will be generated, which will lead to the following result:

```
product = { [IED1, IED1, IED1], [IED1, IED1, IED2], ..., [IED3,IED3,IED3] }
```

Then the index will be used in order to create the mapping, which results in:

[3] https://www.oracle.com/technical-resources/articles/javase/jaxb.html

Algorithm 1 Mapping algorithm

```
1: system := parse_and_load("system.xml")
2: scl := parse_and_load("scl.xml")
3: solutions := map(bay : ∅ for all bay in set(get_bays(scl)))
4:
5: for all bay in set(get_bays(scl)) do
6:     ieds := ordered_set(get_ieds(scl, bay))
7:     fbs := ordered_set(get_fbs(system))
8:
9:     product := cartesian_power( ieds, length(fbs) )
10:    solution_space := set(map_by_index(val, fbs) for all val in product)
11:
12:    for all solution in solution_space do
13:        for all ied in keyset(solution) do
14:            for all fb in solution[ied] do
15:                if not check_if_ied_supports_fb(system, ied, fb) then
16:                    reduce_solution_space(solution_space, ied, fb)
17:                    break and continue with next solution
18:                end if
19:                                    ▷ check_fb_level_constraints(fb, solution)
20:            end for
21:
22:            if not check_device_level_constraints(ied, solution) then
23:                reduce_solution_space(solution_space, solution)
24:                break and continue with next solution
25:            end if
26:        end for
27:
28:        for all connection in set(get_connections(system)) do
29:            mapping := get_connection_map(solution, connection)
30:
31:            if not check_if_ieds_can_communicate(scl, solution, mapping)
   then
32:                reduce_solution_space(solution_space, mapping)
33:                break and continue with next solution
34:            end if
35:                ▷ check_connection_level_constraints(connection, solution)
36:        end for
37:
38:            ▷ check_application_level_constraints(bay, ieds, fbs, solution)
39:
40:        calculate_factor(solution)
41:    end for
42:                        ▷ found solutions for bay if solution_space not empty
43:    solutions[bay] := solution_space
44: end for
```

```
solution_space = { ..., { IED1 → [FB1], IED2 → [FB2, FB3]}, ...  }
```

The solution space then contains all possible combinations. Afterward, the idea is to reduce the size of this solution space successively. This idea can be achieved by examining each solution one by one and checking whether they are valid or not. To increase efficiency, if one solution is invalid, it is advisable to remove all other 'similar' solutions from the solution space. The meaning of 'similar' in this context depends on why a solution is invalid and will be explained later on in more detail. Currently, a solution is considered valid if it meets three conditions: 1. IEDs must support the mapped FBs, 2. the constraints must be compliant with, and 3. IEDs must be able to communicate with each other according to their mapped FBs.

Algorithm 2 shows the check whether an IED supports an FB or not. Firstly, a distinction is made whether the FB is an IEC 61850 FB, i.e., an FB representing an LN or another FB like an IEC 61499 FB. This information is located in the identification field of the FB type. In IEC 61850 FBs, a check if the IED supports this LN has to be done. If so, it has to be checked whether all DOs used are supported. These are DOs, which are the start or endpoint of a connection. In future versions, the DAs contained in the DOs must also be checked, as there are optional ones here. However, this check is a bit more complicated, so it is excluded from this pseudo code. If one of these checks fails, it is clear that it is impossible to map this FB to this IED, and therefore the solution can be discarded from the solution space. However, the solution space can be reduced even further by removing all solutions in which this mapping also occurs. This reduction is made at line 16 of Algorithm 1.

After checking whether all FBs of the IEDs of a possible solution are supported, the next step is to check whether the IED level constraints are met. The format of these has already been discussed in the constraint database section. There are two different types of constraints, asserting and accumulating constraints. An asserting constraint will be satisfied if the device and the FB check succeed. In the case of accumulating constraints, it is a bit more complicated. Here the individual values of the FBs checks are added up and stored in the accumulating store. The accumulated value is then compared with the device-level check value. If the accumulated value is higher than the device-level check value, the constraint is not satisfied. However, it should be noted, that also the communication FBs and other generated FBs have to be included here. Algorithm 3 illustrates this device-level constraint check. Here too, it is possible to reduce the solution space if the result is negative. All solutions with the same FB mapping for the corresponding IED or in which this FB mapping is a subset can be removed .

Finally, the last check concerns communication. It must be checked whether the mapped IEDs of connected FBs can communicate with each other. This check is also presented here in a simplified form. The version only checks whether both IEDs are in the same subnet. One would have to consider router and switch functionality as well as the existence of servers. Besides, a router only forwards specific messages, which means that the DOs used must also be taken into account. Algorithm 4 shows this simplified form. Here, any solution where FB a is mapped to IED A and FB b to IED B can

Algorithm 2 Check if IED supports FB

```
1:  function CHECK_IF_IED_SUPPORTS_FB(system, ied, fb)
2:      fb_type := get_type_from_library(fb.type)
3:
4:      if is_iec_61850_fb(fb_type) then
5:          lns := set(get_supported_LNs(ied))
6:
7:          if get_name(fb) in lns then
8:              for all connection in set(get_fb_connections(system, fb)) do
9:                  dos := get_used_data_object(fb, connection)
10:
11:                 if not check_if_ied_supports_data_objects(ied, dos) then
12:                     return false
13:                 end if
14:             end for
15:             return true
16:         end if
17:         return false
18:     end if
19:     return true
20: end function
```

be removed in case of a negative result. Hereby, FB *a* refers to the FB from which the connection starts, and FB *b* refers to the FB to which the connection ends. In Algorithm 1, these two mappings are obtained via `get_connection_map(solution, connection)` (see line 29).

If all checks are positive, a valid solution has been found for this bay. Next, a factor is calculated, which provides a rating for the solution. This factor may be derived from one specific or a combination of multiple optimization criteria and is intended to assist the Substation Maintainer in selecting the most suitable solution for deployment to the devices. Algorithm 5 shows the implementation of a simple function, which is applied to the found solutions. Here, the factor represents the minimal percentage memory utilization, i.e., a key indicator that shows how well the FBs were distributed. Of course, more sophisticated calculations can be implemented for productive use.

In Algorithm 1, there are also comments indicating where additional constraint checks have to be placed. For example, FB level constraint checks should be added at line 19, while line 35 is suitable for connection-level constraint checks. Their structure will be analogue to the shown device level constraint check. The solutions found are then stored in the Possible Mappings. Its structure is described in more detail in the next section.

Algorithm 3 Check device level constraints

```
1:  function CHECK_DEVICE_LEVEL_CONSTRAINTS(ied, solution)
2:      acc_store := ∅
3:
4:      for all fb in include_generated_Fbs(solution[ied]) do
5:          for all constraint in get_device_level_constraints() do
6:              if get_type(constraint) == "Asserting" then
7:                  if not device_check(constraint, ied) then
8:                      return false
9:                  else if not fb_check(constraint, fb) then
10:                     return false
11:                 end if
12:             else if get_type(constraint) == "Accumulating" then
13:                 acc_store[constraint] += fb_check(constraint, fb)
14:             end if
15:         end for
16:     end for
17:
18:     for all constraint in get_device_level_constraints() do
19:         if get_type(constraint) == "Accumulating") then
20:             if acc_store[constraint] > device_check(constraint, ied) then
21:                 return false
22:             end if
23:         end if
24:     end for
25:
26:     return true
27: end function
```

4.8 Possible Mappings Repository

The Device Mapper returns the found possible mappings as a result. These are saved in an XML file in this framework. Figure 4.21 shows the location and connections of this component within this framework.

Figure 4.22 shows the corresponding structure used in this framework, while the XSD can be found in Appendix 7.1.2. Besides the used system and application, a corresponding XML file contains all found mappings. These mappings are identified by a Universally Unique Identifier (UUID). Additional to an optional factor that provides a rating for the solution based on optimization criteria, it contains the mapping of the individual FBs to the different IEDs.

Algorithm 4 Check if IEDs can communicate

```
1: function CHECK_IF_IEDS_CAN_COMMUNICATE(scl, solution, mapping)
2:     for all subnetwork in set(get_subnetworks(scl)) do
3:         found := set( (ied → false) foreach ied in keyset(mapping))
4:
5:         for all accesspoint in set(get_accesspoints(scl, subnetwork))) do
6:             if accesspoint.Ied in keyset(mapping) then
7:                 found[accesspoint.Ied] = true
8:             end if
9:
10:            if found[ied] == true foreach ied in keyset(mapping) then
11:                return true
12:            end if
13:        end for
14:    end for
15:    return false
16: end function
```

Algorithm 5 Example factor function

```
1: function CALCULATE_FACTOR(solution)
2:     factor := 1
3:
4:     for all ied in keyset(solution) do
5:         free_space := get_total_space(ied)
6:
7:         for all fb in include_generated_Fbs(solution[ied]) do
8:             free_space −= get_size(fb)
9:         end for
10:
11:        percentage := free_space/ get_total_space(ied)
12:
13:        if percentage < factor then
14:            factor := percentage
15:        end if
16:    end for
17:
18:    return factor
19: end function
```

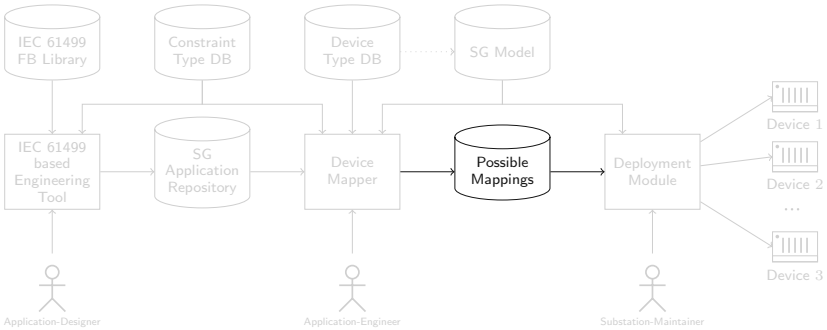


Figure 4.21: Location of the Possible Mappings within the Framework

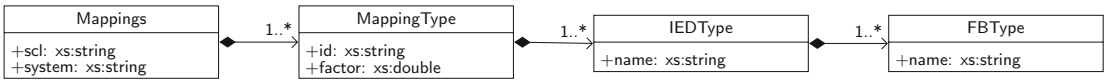


Figure 4.22: Mapping Files XML Structure

Figure 4.2 illustrates this with a small excerpt from a sample XML file. This file serves as an input to the deployment component.

```
<Mappings SCL="scl_1.xml" System="system_1.xml">
  <Mapping id="CAFECAFE-CAFE-CAFE-CAFE-012345678910" factor="0.69">
    <IED name="IED_1">
      <FB name="FB_2"/>
      <FB name="FB_3"/>
    </IED>
    <IED name="IED_2">
      <FB name="FB_1"/>
    </IED>
  </Mapping>
  <!-- ... -->
</Mappings>
```

Listing 4.2: Example Mappings File

4.9 Deployment Module

Based on the Device Mapper's output, a substation maintainer can select a suitable mapping by its ID and can further optionally adapt it if wished. Subsequently, the deployment module updates the result accordingly in the two main files, i.e., the SCL and the System XML files. Figure 4.23 shows the location and connections of this component within this framework.

Algorithm 6 shows the generation of the system file in a simplified way. After selecting the mapping by its ID, the template device, which was added in order to guarantee standard conformance (see Section 4.3) has to be removed. For each IED, a device and

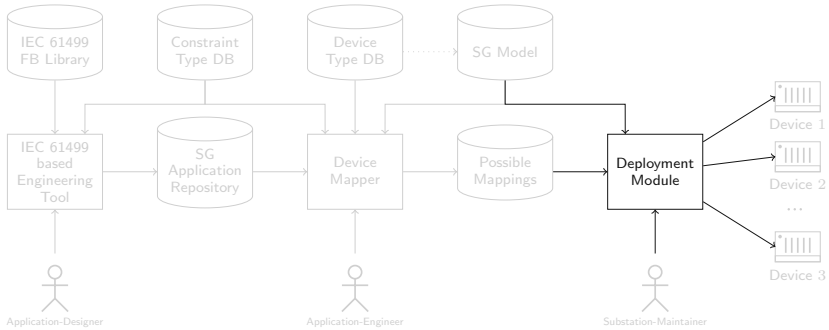


Figure 4.23: Location of the Deployment Module within the Framework

its resource is created. Furthermore, an `E_RESTART` FB is used to initialize the FBs. Each initializable FB, i.e., FBs with the event INIT and each startable FB, i.e., FBs with the event START, are connected to `E_RESTART`. IEC 61850 FBs are not created, since they represent an IEC 61850 device and are hence only published to and subscribed from. One can think of them as a subapplication that contains a publish and subscribe FB.

For each connection, it will then be checked whether the connected FB is also mapped to the same device or not. If not, a subscribe or publish FB has to be generated and connected accordingly. Depending on the FB type, this is either a multicast or a GOOSE subscription or publish FB. Figure 4.24 shows how these FBs look like in the case of multicasts. Here, ID represents the multicast address and SD_x and RD_x represents the value(s) to be written or read, where the maximum x depends on the FB. GOOSE subscriber and publisher FBs almost look the same, but require a Media Access Control (MAC) and an Internet Protocol (IP) address instead.

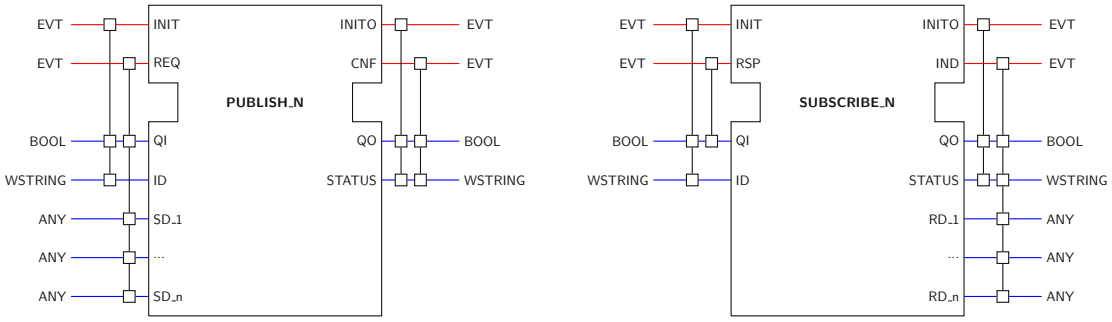


Figure 4.24: IEC 61499 Publish/Subscribe Function Blocks

A system file can then either be used to configure the individual devices using Integrated Development Environments (IDEs) such as 4DIAC or to generate a script that configures them directly using management commands from IEC 61499. Especially when using 4DiacForte, the management commands can be passed as a text file on startup. These text files are called Forte Boot Files and will also be produced by this framework. They contain comma separated values, where the first entry of each line describes the affected

Algorithm 6 Generate System File

```
1:  function GENERATE_SYSTEM(system, mappings, id)
2:      mapping := get_mapping_by_id(mappings, id)
3:      remove_template_device(system)
4:      create_segment(ETHERNET)
5:
6:      for all ied in keyset(mappings) do
7:          resource := add_device_and_resource(get_name(ied))
8:          add_fb(system, resource, E_RESTART)
9:          connect_to_segment(ied, ETHERNET)
10:
11:         for all fb in mapping[ied] do
12:             if is_iec_61850_fb(fb) then
13:                 continue
14:             end if
15:
16:             if is_initializable(fb) or is_startable(fb) then
17:                 connect_to_restart(system, resource, fb)
18:             end if
19:
20:             for all con in get_connections(system, fb) do
21:                 if get_from(con) == fb then
22:                     if get_to(con) in mapping[ied] then
23:                         add_connection(system, resource, fb, con)
24:                     else
25:                         connect_to_publish(system, resource, fb, con)
26:                     end if
27:                 else
28:                     if get_from(con) in mapping[ied] then
29:                         add_connection(system, resource, fb, con)
30:                     else
31:                         connect_to_subscribe(system, resource, fb, con)
32:                     end if
33:                 end if
34:             end for
35:         end for
36:     end for
37:     return system
38: end function
```

resource and the second entry describes the corresponding action using XML. This action can either be CREATE, WRITE, START, or STOP. Using these commands it is possible to create FBs, connect them, and start the application. Figure 4.25 shows the corresponding structure used in this framework, while the XSD can be found in Appendix 7.1.3.

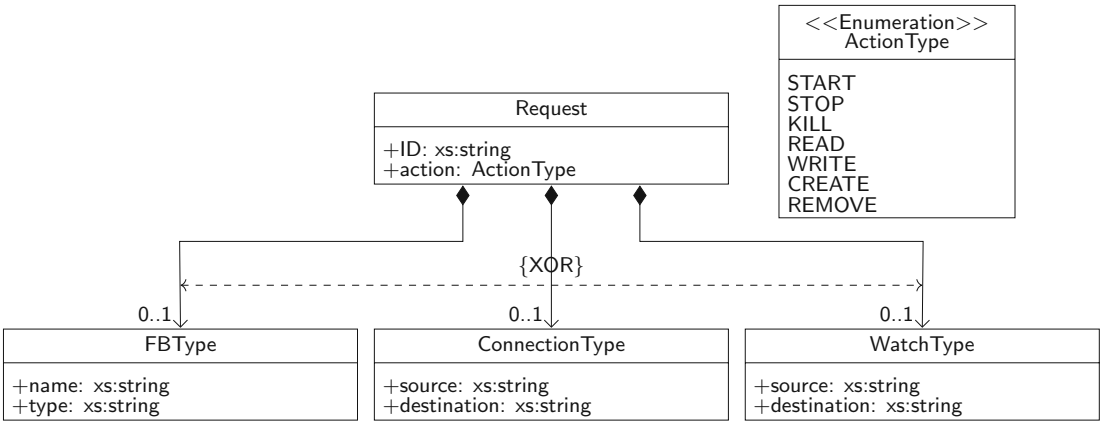


Figure 4.25: Forte Boot Files XML Structure

In the SCL file, the subscriptions have to be set up according to the selected mapping. This can be done by iterating the connections like before, and adding ExtRef tags to the inputs of the concerned IEDs. Afterward, it is easy to generate the individual CID files by copying the relevant sections, since they represent only a subset of the SCL file. The CID files can then be used to directly configure the IEDs.

4.10 Deployment Files

Besides the adapted system and SCL files, the deployment module finally provides CID files for each device and a system XML file for configuring the IEC 61499 runtime environments of the devices. Furthermore, Forte Boot Files are generated. After the Substation Maintainer imports the corresponding files to the devices, the newly developed application will run in the Substation.

CHAPTER 5

Use-Case Implementation

This chapter describes the approach in detail by implementing a specific use-case. An important example in the SG area is the FLISR scenario. FLISR ensures the self-healing capabilities of an SG and, thus, increases robustness. As the name suggests, it is a multi-stage approach. The first step is to find the faulty feeder or section. This section is the portion of a feeder that is located between two switches/reclosers. In a further step, this area will then be isolated from its adjacent switches. Finally, the areas that are no longer supplied must be supplied by (other) feeders. It requires the corresponding normally open tie switches to be closed. However, it should be noted that only feeders that still have sufficient capacity can be used. [PVM13]

5.1 Application Modelling

For a prototype, an attempt is being made to implement the first part of the FLISR scenario. Figure 5.1 shows the modeled application. MMXU represents a Grid Measurement Device (GMD), PIOC represents an overcurrent detection device, PTRC represents a protection trip conditioning function, and XCBR represents a circuit breaker. Additionally, there exists an IEC 61499 FB STORE_IN_DB. The application works like this: As soon as MMXU reads a new value, an event is triggered. The overcurrent detection function then checks whether the received value will be above the threshold set via StrVal and if so, it also triggers an event. CDCs are represented by structured data types which can be composed using COMPOSE FBs. As described in detail in the Appendix, the CDC ASG (Analogue Settings) consists of the attributes setMag and units, which correspond to the types Vector and Unit. To specify a threshold of 12 A, the vector's magnitude attribute f is set to 12, and amperes are selected as the unit with multiplier 1. PTRC then bundles the output of one or more protection functions and sends the trip to a circuit breaker, which opens the circuit. Besides, the detected fault is written to a database, represented by the STORE_IN_DB FB. This FB is assumed to have been taken from an existing

library. In the example application the database server runs on port 5000 of a device with the IP address 10.0.0.30.

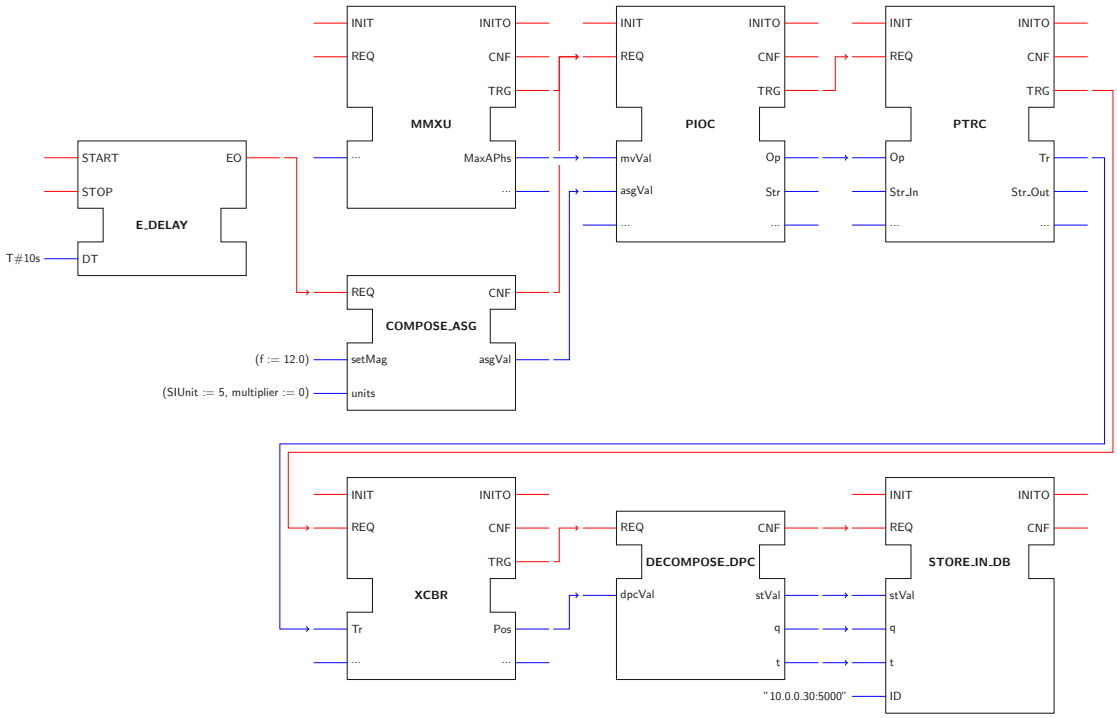


Figure 5.1: IEC 61850 Example Application Part 1

Additionally, there is another IEC 61499 application shown in Figure 5.2. Here, it is read periodically (E_CYCLE), in this case, every 10 s, from the database (READ_FROM_DB) whether an error has occurred. If this is the case, a message will be generated, which will then be forwarded to the SEND_SMS FB. This FB ensures that technicians are informed of potential problems via Short Message Service (SMS).

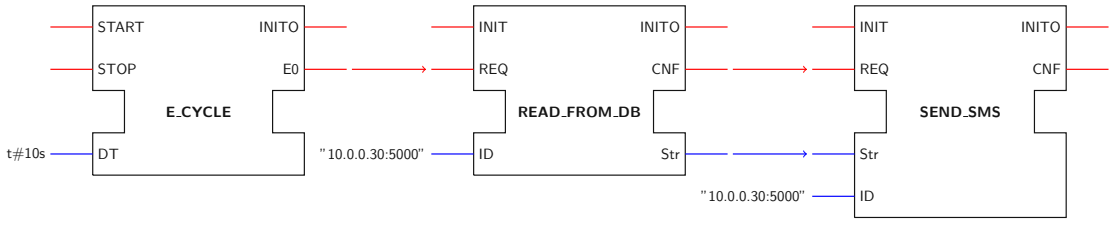


Figure 5.2: IEC 61850 Example Application Part 2

The application XML file that has been produced by the IEC 61499 based engineering tool can be found in Appendix 7.5.2. This file will then be used by the device-mapper

5.2 Constraints

As indicated in the previous chapter, two constraints are used. Listing 5.1 shows the corresponding XML file.

```
<?xml version="1.0" encoding="UTF-8"?>
<Constraints>
   <DeviceConstraint
      name="freeSpaceConstraint"
      type="accumulating"
      level="device-level"
      device-check="IED.Private('Memory')"
      fb-check="Size(FB)"
   />
   <DeviceConstraint
      name="iec61499ComplianceConstraint"
      type="asserting"
      level="device-level"
      device-check="IED.Private('Programmable') == True"
      fb-check="ref(FB.Type).Identification.Standard == 'IEC 61499'"
   />
</Constraints>
```

Listing 5.1: Constraints

The freeSpaceConstraint checks whether there is enough memory on the devices by adding up the sizes of the FBs and comparing them to the available memory. This information can be found in the private field memory. The second constraint is then responsible for checking whether IEC 61499 FBs can be distributed to the respective device. This is only possible if the device has been marked as 'programmable'. The respective FB type, i.e. whether IEC 61499 or IEC 61850 FBs are involved, can be checked via its standard field.

5.3 Example Substation

As mentioned in Chapter 2, IEC 61850 already provides a standardized description of existing grids in the form of single-line diagrams. It can be done using SSD files, which are a subset of SCL files. Figure 5.3 shows the graphical representation of such a grid, while the respective SSD file included in the complete SCL file can be found in Appendix 7.5.1. In the example, there are three low voltage feeders (in Voltage Level L), supplied by three transformers (T1, T2, and T3), which are connected to a medium voltage network (Voltage Level M). The feeders are all connected to the transformers via a busbar and have the same structure. On top of each feeder is a circuit breaker (QA), which allows switching even in a fault current situation. Besides, each feeder, has two more circuit switches (QB1 and QB2). As both the feeders and the busbars are located in their bays, these elements can be uniquely identified (e.g., L/Q7/QA for the circuit breaker in feeder 1 in Voltage Level L). This grid also has two so-called tie switches (QB), which connect feeders 1 and 2, or feeders 2 and 3, in order to be able to ensure an alternative supply for sections not affected in the event of a fault. Thus, each feeder consists of three sections,

section 1 between QA and QB1, section 2 between QB1 and QB2, and section 3 below QB2.

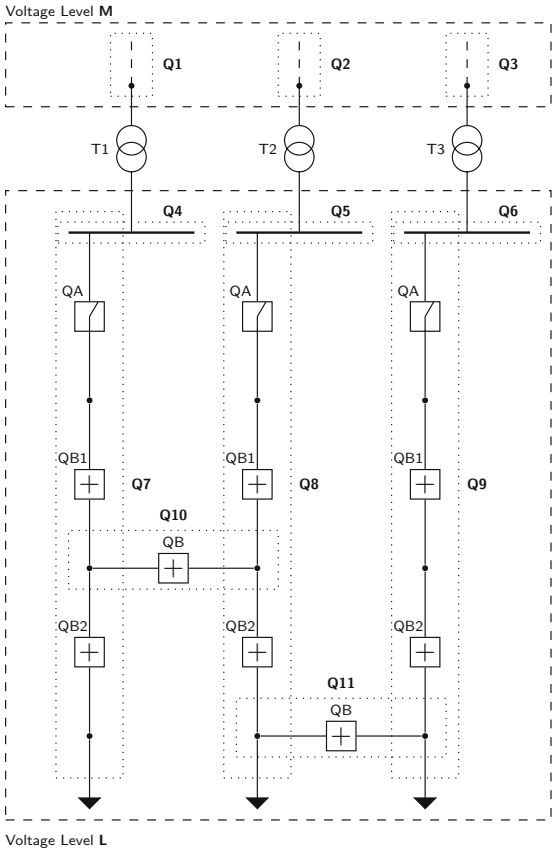


Figure 5.3: IEC 61850 Example Substation

After describing the existing grid topology, the next step is to describe the existing infrastructure, i.e., the existing devices and their interconnection. For this purpose, reduced SCD files are used. Figure 5.4 shows the schematic representation of the communication infrastructure of Bay Q7, while again, the description included in the SCL file can be found in Appendix 7.5.1.

Table 5.1 shows which and how many instances of LNs are supported by the IEDs and whether the respective IEDs are programmable. For the sake of simplicity, the table contains only LNs used in the use-case application. Of course, this information can also be found in the complete SCL file in Appendix 7.5.1.

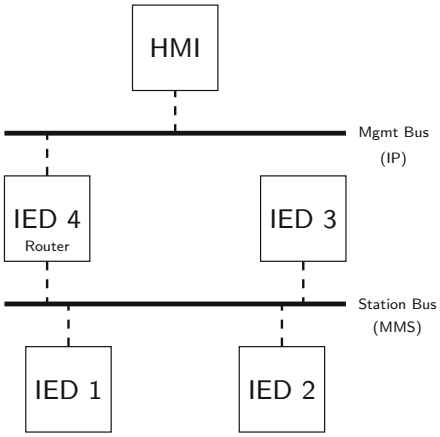


Figure 5.4: IEC 61850 - Example Communication Infrastructure (Bay Q7)

	IED 1	IED 2	IED 3	IED 4
Programmable (Total Space)	No	No	Yes (1000 kB)	Yes (1000 kB)
MMXU LN (#Instances)	Yes (1)	Yes (1)	No	No
PIOC LN (#Instances)	No	Yes (1)	Yes (1)	Yes (1)
PTRC LN (#Instances)	No	No	Yes (1)	No
XCBR LN (#Instances)	Yes (1)	No	No	No

Table 5.1: IEC 61850 - Example IED-LN Mapping

5.4 Device Mapping

Using the IEC 61499 based Engineering Tool's output together with the previously described substation, it is possible to use the device mapping algorithm described in the previous chapter. The algorithm will be demonstrated by applying it to the Bay Q7. Assuming all sets are processed alphabetically, and all IEC 61499 FBs need about 100kB of space, the algorithm finds the following first possible solution:

- **IED 1**: MMXU FB, XCBR FB
- **IED 2**: PIOC FB
- **IED 3**: PTRC FB, E_CYCLE FB, READ_FROM_DB FB, SEND_SMS FB, STORE_IN_DB
- **IED 4**: COMPOSE_ASG FB, DECOMPOSE_DPC FB, E_DELAY FB

Assuming that the available memory on IEDs 3 and 4 is reduced from 1 MB to 500 kB, it is easy to see that the algorithm also recognizes if there are no valid solutions for mapping, and, thus, the solution space becomes empty without finding a solution. It

is also interesting to look at the size of the solution space during the mapping process. Assuming that the FBs and IEDs are processed alphabetically, in the first possibility all FBs are mapped to IED 1. The size of the solution space in the beginning is calculated as follows:

$$|solution\_space| = \#IEDs^{\#FBs} = 4^{11} = 4194304$$

Since IED 1 does not support the COMPOSE_ASG FB, for example, all solutions that have this mapping are removed from the solution space so that after this first 'round', the size of the solution space is:

$$|solution\_space| = 4^{11} - \frac{1}{4}4^{11} = \frac{3}{4} \cdot 4^{11} = 3 \cdot 4^{10} = 3145728$$

In the next solution then the COMPOSE_ASG FB is mapped to IED 2, while all other FBs are still mapped to IED 1. Here, for example, IED 2 does not support the DECOMPOSE_DPC FB. Thus, the solution space is reduced by a quarter again minus the mappings where both FB (COMPOSE_ASG and DECOMPOSE_DPC) are mapped to IED 1, i.e.

$$|solution\_space| = 4^{11} - \frac{1}{4}4^{11} - \frac{1}{4}4^{11} + \frac{1}{16}4^{11} = 2359296$$

Using the inclusion-exclusion-principle, it is easy to calculate the remaining solutions. After ten rounds the size of the solution space will already be reduced by about 200 times, to 209952. It can be seen that if only a few devices support certain FBs, the solution space can be reduced quite quickly. Even impossible connections can reduce it further. However, many multi-purpose devices with strict constraints slow down the process because they can only reduce the solution space one by one. Figure 5.5 shows the size of the solution space over the number of iterations for the given example.

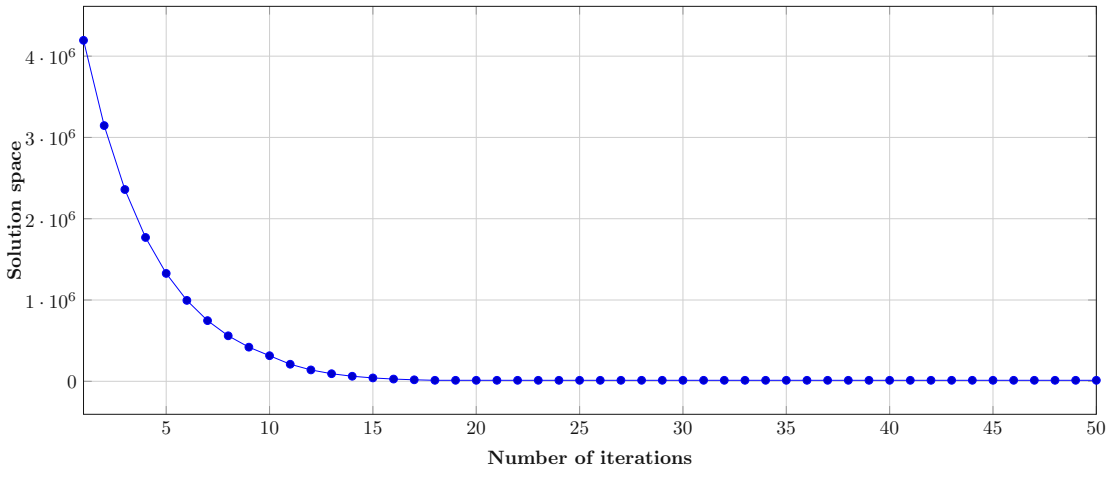


Figure 5.5: Decreasing size of the solution space over number of iterations

In this context, the question arises which is the best solution, or is there a best solution. The answer is, it depends on the particular requirements. For example, one requirement may be to use devices from as few different manufacturers as possible. In the next chapter, possible extensions are described in more detail. In the current realization of the framework, the minimal percentage memory utilization is used. Using this factor, the following solution would be a best solution and will be used for the deployment:

- **IED 1**: MMXU FB, XCBR FB
- **IED 2**: PIOC FB
- **IED 3**: E_CYCLE FB, PTRC FB, READ_FROM_DB FB, STORE_IN_DB
- **IED 4**: COMPOSE_ASG FB, DECOMPOSE_DPC FB, E_DELAY FB, SEND_SMS FB

However, it should be noted that the automatically created publish and subscribe FBs as well as the always existing E_RESTART FB are not shown here, but still considered. An extract of the output of the device-mapper under the aforementioned conditions can be found in Appendix 7.6.1.

5.5 Deployment

The generated mapping file together with the chosen application and substation can then be used to create the deployment files. Assume the substation engineer tries to deploy the 'best' solution found before. Two things have to be done then: first of all, depending on the application and the mapping, the complete SCL file has to be split up to CID files, including the corresponding GOOSE subscriptions for all devices. Secondly, in the system XML file, the devices need to be created. Additionally, the publisher and subscriber FBs and as well as initialization FBs will be created. These files can be found in the Appendix. Figures 5.8 and 5.9 show the portion of the application mapped to IED 3 and IED 4, respectively. Here, the publish and subscribe FBs are used for communicating with other IEC 61499 deviced, while the pub_goose and the sub_goose FBs are used for communicating with the IEC 61850 devices. Furthermore, it can be seen, that all FBs are initialized by the E_RESTART FBs of the two devices.

Figure 5.6 then shows an example activity diagram of the application. Succeeding a delay of 10 s the COMPOSE_ASG FB on IED 3 configures the threshold of the PIOC algorithm on IED 2. After IED 1 has published a MMXU measurement greater than the configured threshold, PIOC on IED 2 will publish an operate command which will be forwarded by the PTRC. This leads the Circuit Breaker to switch its position. IED 4 then publishes the decomposed positional value, which will be stored by IED 3 in the database running on the host. IED 3 then reads and publish the read value, which is sent to the technician via SMS by IED 4.

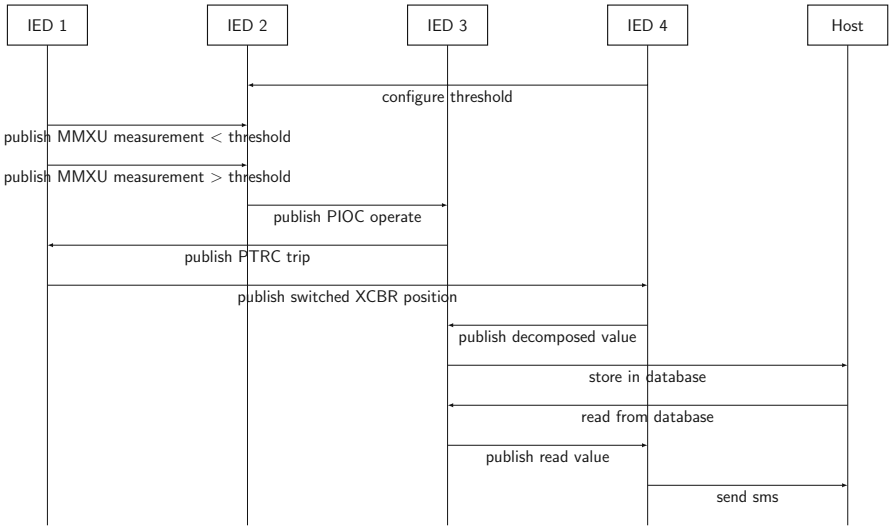


Figure 5.6: Example Activity Diagram of the distributed Application

5.6 Testbed

As a proof of concept this application has been deployed to a simple testbed. Figure 5.7 shows the used network structure. All devices have a fixed IP and MAC address in this setup. Besides libIEC61860, IEDs 3 and 4, which are programmable, additionally run 4DiacForte. The Host runs a Python script, which represents the Database as well as the SMS server. Using a bridged network the devices were then deployed using Docker. The corresponding docker-compose file can be found in Appendix 7.7.1.

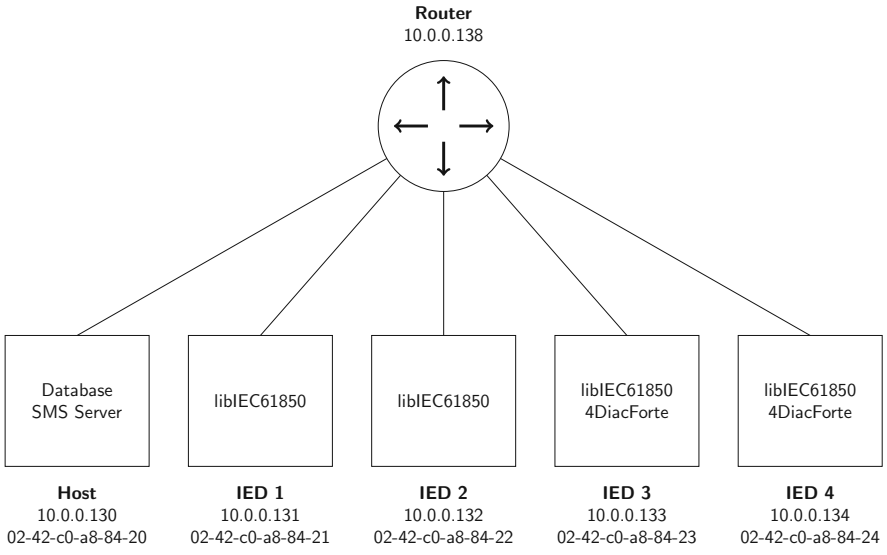


Figure 5.7: Example Network Structure used in the Testbed

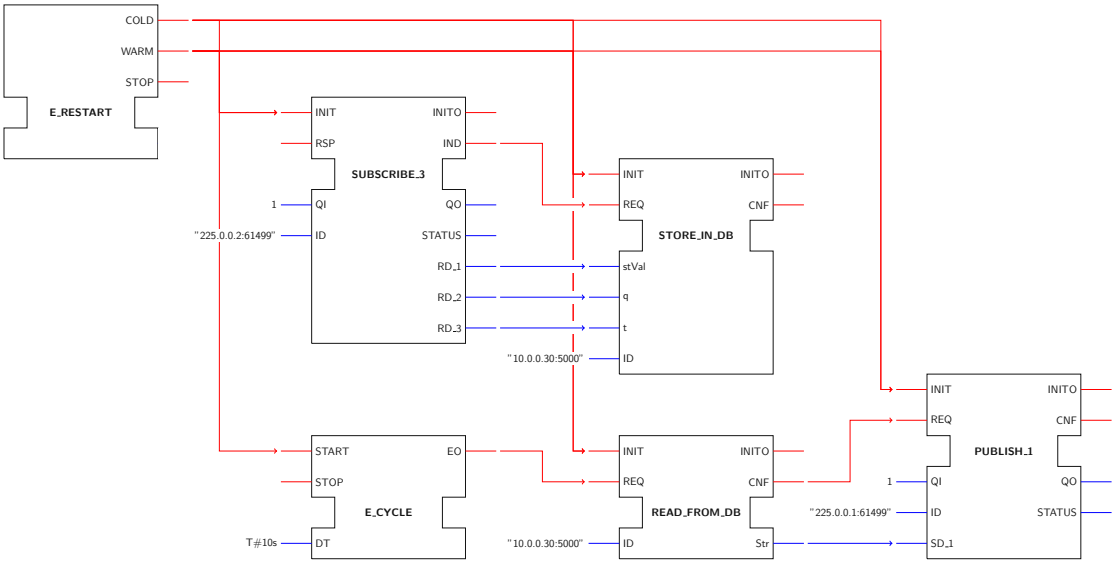


Figure 5.8: Function Blocks mapped to IED 3

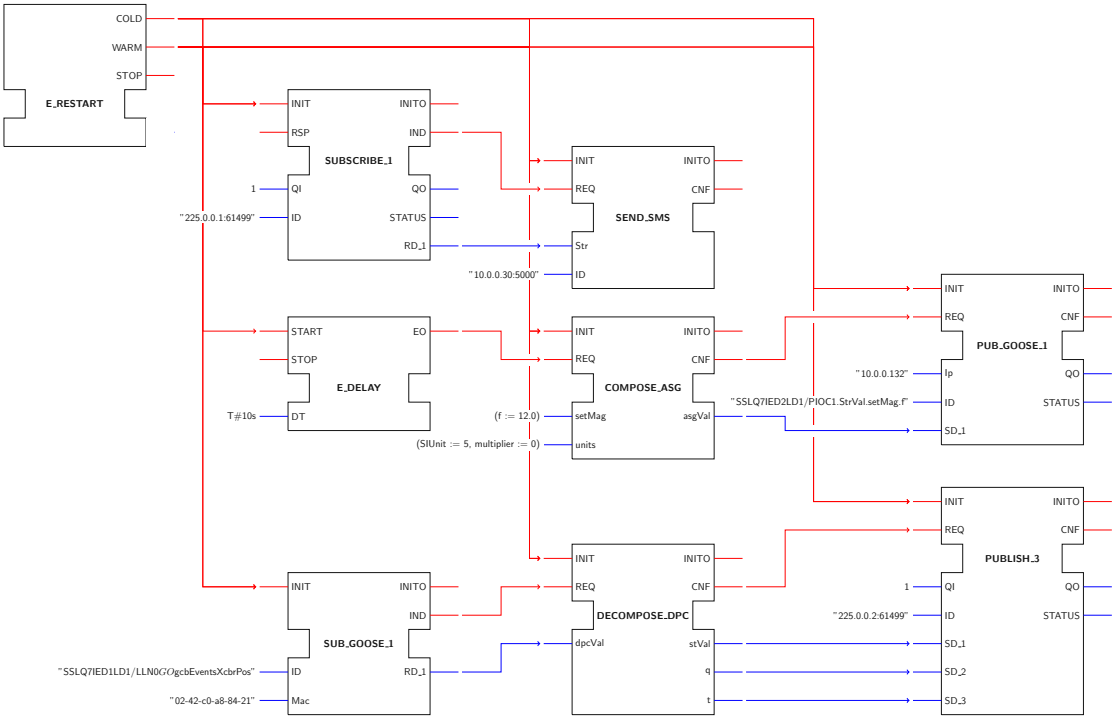


Figure 5.9: Function Blocks mapped to IED 4

CHAPTER 6

Conclusion and Outlook

With the increasing pervasion of the electrical grid with smart devices, fundamental changes in the way energy is generated and transported are not only becoming possible but are already implemented in today's grids. This, for example, not only enables decentralized power generation by end customers, but also provides benefits for them like dynamic pricing. However, this also implies an increasing complexity of application development. This thesis deals with exactly this issue.

While traditional approaches for the development of new applications are usually applied to a specific grid, this framework describes possibilities of a generic application development. For this purpose, FBs from IEC 61499 are utilized. In a nutshell, this is the central innovation of this work. Besides traditional engineering methods like the SGAM model and IEC 62559, new approaches like the model driven approach PSAL have been investigated. Although PSAL also tries to deal with the increasing complexity and accelerates the development process, it still presupposes an existing SG.

After studying other approaches from the SG domain as well as from other domains, the basic structure of the proposed framework was determined. Three modules were identified that such a framework must have, namely, a module for application modeling, a module to map the application to the devices and a module to deploy the distributed application then. Subsequently, the individual requirements for these modules were identified and different ways of implementing them were proposed.

In summary, the application designer first develops an application with the help of FBs. Two types of function blocks are available: IEC 61850 FBs, which represent SG functionality, and IEC 61499 FBs. An application engineer then starts the mapping process by selecting a specific application and a specific SG. After optional changes, deployment files based on a selected mapping are automatically generated, which can be imported by the substation maintainer.

Based on a specific use-case, the first part of the FLISR scenario, the correct behavior of the application was verified and demonstrated in a test environment.

Some ideas for the framework were not discussed in such detail for reasons of simplicity and are thus considered a future work. So, for example, an approach was chosen where one FB represents exactly one node. As already mentioned, the application modeling can be made even more powerful by softening this requirement. However, this leads to an increase of the implementation complexity. Nevertheless, it would be an interesting starting point for further investigations. Furthermore, the proof-of-concept is currently limited to reading and writing values of IEC 61850 LNs, but it is nevertheless possible to extend the FBs so that other services from IEC 61850 are also supported, such as reading and writing log files. This extension could either be done by introducing an additional event or an additional data input and output, which, combined with the REQ event, triggers the appropriate action.

Also, one could develop an integrated tool similar to the AUTERAS software for productive use. While this work focuses on the development of new SG applications, it would also be interesting to evaluate how a domain-independent functional description could look like, or how this framework could be transformed to other domains.

There are also some possible adaptations to the existing framework. In the existing framework the device database is only used implicitly. The reason that it is still included is to be able to propose (best) solutions in a future work. Here different gradations are conceivable. As in this framework, devices could be predefined in the existing grid and only if the mapping does not find a solution, additional devices could be suggested. However, it is also conceivable that the framework proposes a completely new device infrastructure. However, this comes at the price of a significantly higher complexity of finding solutions, i.e. the solution space is increased remarkably. Especially here, different heuristics could be applied and evaluated, which were not the focus of this thesis. Furthermore, this database can also be used for the generation of FBs. As mentioned, the IEC 61850 standard only defines a data model. The concrete inputs required by the various algorithms depend on the manufacturer's algorithm. However, common inputs can be evaluated using this database. Or alternatively, an external knowledge graph could be filled with this information.

The mapping algorithm itself can also be further improved. It has already been mentioned that a simplified check of the IEC 61850 FB connectivity is performed in this framework. Theoretically, it has also to be checked whether optional DOs and DAs are present. Likewise, it must be checked whether these are actually accessible via an IECs 6850 server. Other communication relationships also have to be considered in more detail. For example, not all data is necessarily transported via GOOSEs, but some data points are transported via GSSEs and SMVs. Also merging units could be considered.

The constraint database is currently very simple. Also a simple optimization algorithm is used at the moment. This should be made more flexible and extensive in future versions. Especially if the structure and the algorithms become more complex, it would also be

desirable to automatically validate the result of the mapping, as well as the result of the deployment.

In this thesis, the 4DiacForte runtime environment is used in the test bed. As already discussed, the FBs used must already be compiled together with this runtime environment. Here, it would be interesting which other possibilities there are to store the code directly in the FBs. One problem with this is that the IEC 61499 standard does not provide any information on how the embedded code should be executed. For example, the LUA Engine for 4DiacForte could be tested, which allows including a source code, even though it is only limited to this specific runtime environments. In general, other runtime environments could be tested using the given framework.

CHAPTER 7

Appendix

7.1 Framework Schemata

7.1.1 Constraint Schema

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Constraints">
  <xs:complexType>
     <xs:sequence>
      <xs:choice maxOccurs="unbounded">
        <xs:element type="DeviceLevelConstraint" name="DeviceConstraint"/>
        <xs:element type="FbLevelConstraint" name="FbConstraint"/>
        <!-- ... -->
      </xs:choice>
     </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:simpleType name="ConstraintType">
   <xs:restriction base="xs:string">
     <xs:enumeration value="accumulating" />
     <xs:enumeration value="asserting" />
   </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="ConstraintLevel">
   <xs:restriction base="xs:string">
     <xs:enumeration value="application-level" />
     <xs:enumeration value="device-level" />
     <xs:enumeration value="fb-level" />
     <xs:enumeration value="connection-level" />
   </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="DeviceLevelConstraint">
   <xs:attribute name="name" type="xs:string" use="required"/>
```

```
    <xs:attribute name="type" type="ConstraintType" use="required"/>
    <xs:attribute name="level" type="ConstraintLevel" use="required"
              fixed="device-level"/>
    <xs:attribute name="device-check" type="xs:string" use="required" />
    <xs:attribute name="fb-check" type="xs:string" use="required" />
  </xs:complexType>
  <xs:complexType name="FbLevelConstraint">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="type" type="ConstraintType" use="required"/>
    <xs:attribute name="level" type="ConstraintLevel" use="required"
              fixed="fb-level"/>
    <xs:attribute name="fb-check" type="xs:string" use="required" />
    <xs:attribute name="fb-check-acc" type="xs:string" use="optional" />
  </xs:complexType>
</xs:schema>
```

Listing 7.1: Constraints XML Schema

7.1.2 Mappings File Schema

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Mappings">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="MappingType" name="Mapping" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="SCL" type="xs:string" use="required"/>
      <xs:attribute name="System" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="MappingType">
    <xs:sequence>
      <xs:element type="IEDType" name="IED" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="factor" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="IEDType">
    <xs:sequence>
      <xs:element type="FBType" name="FB" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

  <xs:complexType name="FBType">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
</xs:schema>
```

Listing 7.2: XML Schema Definition for the Mapping Files

7.1.3 Forte Boot Files Schema

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Request">
    <xs:complexType>
      <xs:attribute name="ID" type="xs:int" use="required"/>
      <xs:attribute name="Action" type="ActionType" use="required"/>
      <xs:sequence>
        <xs:choice minOccurs="0" maxOccurs="1">
          <xs:element type="FBType" name="FB"/>
          <xs:element type="ConnectionType" name="Connection"/>
        </xs:choice>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:simpleType name="ActionType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="START" />
      <xs:enumeration value="STOP" />
      <xs:enumeration value="CREATE" />
      <xs:enumeration value="WRITE" />
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="FBType">
    <xs:attribute name="Name" type="xs:string" use="required"/>
    <xs:attribute name="Type" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="Connectiontype">
    <xs:attribute name="Source" type="xs:string" use="required"/>
    <xs:attribute name="Destination" type="xs:string" use="required"/>
  </xs:complexType>
</xs:schema>
```

Listing 7.3: XML Schema Definition for the Forte Boot Files

7.2 Automatic Generation

7.2.1 IEC 61499 Data Type Generation

This section shows by example of the SPS CDC (see Figure 7.1) how the generation of the IEC 61499 data types is done. First, all used data types are analyzed. In the case of the SPS CDC these are: BOOLEAN, Quality, TimeStamp, VISIBLE STRING 64, VISIBLE STRING 255 and UNICODE STRING 255. The 'primitive' data types can be created directly as DirectlyDerivedType.

The composite data types Quality (see Table 7.3) and TimeStamp (see Table 7.2) must be further decomposed. The TimeStamp type again consists of three fields. While INT32U again can be created directly as DirectlyDerivedType, INT24U is represented as SubrangeType.

SPS - Single Point Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
stVal	BOOLEAN	ST	dchg, dupt	TRUE | FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
units	Unit	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.1: IEC 61850 SPS CDC

TimeStamp type definition			
Attribute name	**Attribute type**	**Value/value range/explaination**	**M/O**
SecondSinceEpoch	INT32U	$(0 \dots MAX)$	M
FractionOfSecond	INT24U		M
TimeQuality	TimeQuality		M

Table 7.2: IEC 61850 Timestamp Type

TimeQuality is again a composite type (see Table 7.4). After the CODED ENUM TimeAccuracy has been created as EnumeratedType, the TimeQuality Type can then be created as Structured Data Type as well as the TimeStamp Type. After the composite data type Quality has also been decomposed according to this procedure, the SPS CDC can be created as Structured Data Type in the last step.

Listing 7.4 shows the resulting XML files, where for the sake of clarity the repeated specification of the DOCTYPE was omitted.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_boolean">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <DirectlyDerivedType BaseType="BOOL"/>
</DataType>

<!-- Other String Types accordingly -->
<DataType Name="iec_61850_visible_string_64">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <DirectlyDerivedType BaseType="STRING(64)"/>
</DataType>
```

Quality type definition			
Attribute name	**Attribute type**	**Value/value range/explaination**	**M/O**
	PACKET LIST		
validity	CODED ENUM	good | invalid | reserved | questionable	M
detailQual	PACKED LIST		M
overflow	BOOLEAN		M
outOfRange	BOOLEAN		M
badReference	BOOLEAN		M
oscillatory	BOOLEAN		M
failure	BOOLEAN		M
oldData	BOOLEAN		M
inconsistent	BOOLEAN		M
inaccurate	BOOLEAN		M
source	CODED ENUM	process | substituted, DEFAULT process	M
test	BOOLEAN	DEFAULT FALSE	M
operatorBlocked	BOOLEAN	DEFAULT FALSE	M

Table 7.3: IEC 61850 Quality Type

TimeQuality type definition			
Attribute name	**Attribute type**	**Value/value range/explaination**	**M/O**
	PACKET LIST		
LeapSecondsKnown	BOOLEAN	$(0 \dots MAX)$	M
ClockFailure	BOOLEAN		M
ClockNotSynchronized	BOOLEAN		M
TimeAccuracy	CODED ENUM		M

Table 7.4: IEC 61850 TimeQuality Type

```
<DataType Name="iec_61850_INT32U">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <DirectlyDerivedType BaseType="UDINT"/>
</DataType>

<DataType Name="iec_61850_INT24U">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <SubrangeType BaseType="UDINT">
      <Subrange LowerLimit="0" UpperLimit="1677215"/>
   </SubrangeType>
</DataType>

<DataType Name="iec_61850_quality_validity">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <EnumeratedType>
      <EnumeratedValue Name="good"/>
      <EnumeratedValue Name="invalid"/>
      <EnumeratedValue Name="reserved"/>
      <EnumeratedValue Name="questionable"/>
   </EnumeratedType>
</DataType>

<DataType Name="iec_61850_quality_detailQual">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <StructuredType>
      <VarDeclaration Name="overflow" Type="iec_61850_boolean"/>
```

```
      <VarDeclaration Name="outOfRange" Type="iec_61850_boolean"/>
      <VarDeclaration Name="badReference" Type="iec_61850_boolean"/>
      <VarDeclaration Name="oscillatory" Type="iec_61850_boolean"/>
      <VarDeclaration Name="failure" Type="iec_61850_boolean"/>
      <VarDeclaration Name="oldData" Type="iec_61850_boolean"/>
      <VarDeclaration Name="inconsistent" Type="iec_61850_boolean"/>
      <VarDeclaration Name="inaccurate" Type="iec_61850_boolean"/>
   </StructuredType>
</DataType>

<DataType Name="iec_61850_quality_source">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <EnumeratedType>
      <EnumeratedValue Name="process"/>
      <EnumeratedValue Name="substituted"/>
   </EnumeratedType>
</DataType>

<DataType Name="iec_61850_quality">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <StructuredType>
      <VarDeclaration Name="validity" Type="iec_61850_quality_validity"/>
      <VarDeclaration Name="detailQual" Type="iec_61850_quality_detailQual"/>
      <VarDeclaration Name="source" Type="iec_61850_quality_source" InitialValue="process"/>
      <VarDeclaration Name="test" Type="iec_61850_boolean" InitialValue="FALSE"/>
      <VarDeclaration Name="operatorBlocked" Type="iec_61850_boolean" InitialValue="FALSE"/>
   </StructuredType>
</DataType>

<DataType Name="iec_61850_timeQuality_timeAccuracy">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <EnumeratedType>
      <EnumeratedValue Name="7" Comment="approx 7.8 ms"/>
      <EnumeratedValue Name="10" Comment="approx 0.9 ms"/>
      <EnumeratedValue Name="14" Comment="approx 61 us"/>
      <EnumeratedValue Name="16" Comment="approx 15 us"/>
      <EnumeratedValue Name="18" Comment="approx 3.8 us"/>
      <EnumeratedValue Name="20" Comment="approx 0.9 us"/>
      <EnumeratedValue Name="31" Comment="unsepecified"/>
   </EnumeratedType>
</DataType>

<DataType Name="iec_61850_timeQuality">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <StructuredType>
      <VarDeclaration Name="leapSecondsKnown" Type="iec_61850_boolean"/>
      <VarDeclaration Name="clockFailure" Type="iec_61850_boolean"/>
      <VarDeclaration Name="clockNotSynchronized" Type="iec_61850_boolean"/>
      <VarDeclaration Name="timeAccuracy" Type="iec_61850_timeQuality_timeAccuracy"/>
   </StructuredType>
</DataType>

<DataType Name="iec_61850_sps">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <StructuredType>
      <VarDeclaration Name="stVal" Type="iec_61850_boolean"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timeStamp"/>
      <VarDeclaration Name="subEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subVal" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subQ" Type="iec_61850_quality"/>
      <VarDeclaration Name="subID" Type="iec_61850_visible_string_64"/>
      <VarDeclaration Name="blkEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.4: Generated IEC 61499 SPS Data Types

7.2.2 IEC 61499 FB Type Generation

This section shows by example of the PIOC LN (see Figure 7.6) how the generation of the IEC 61499 FB types is done. Tables 7.5 and 7.6 shows the structure of the PIOC LN. The automatic generation is done as follows: Based on the LN table, CDCs the FC Settings become input variables, CDCs with the FC Status Information, or measured and metered values become output variables and controls become both input and output variables.

Common LN				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
Descriptions				
NamPlt	LPL	Name Plate		C
Status information				
Beh	ENS	Behaviour		M
Health	ENS	Health		C
Blk	SPS	Dynamic blocking of function described by the LN		O
Controls				
Mod	ENC	Mode		C

Table 7.5: IEC 61850 Common LN

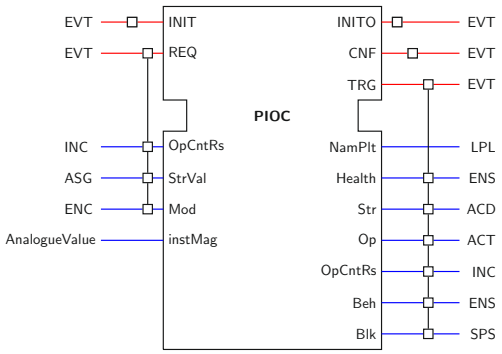
PIOC				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Status information				
Str	ACD	Start		O
Op	ACT	Operate		M
Controls				
OpCntRs	INC	Resettable operation counter		O
Settings				
CBOpCap	ASG	Start value		O

Table 7.6: IEC 61850 PIOC LN

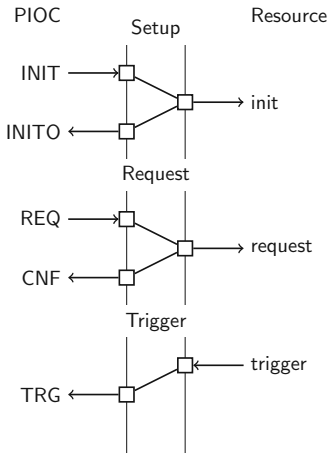
In addition, the input events INIT and REQ, as well as the output events INITO, CNF and TRG are created and connected to the inputs and outputs accordingly, see Figure 7.1a. Figure 7.1b then sketches the corresponding sequence diagrams.

Assuming that the corresponding data types have already been generated, the resulting XML file is shown in Listing 7.5.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="iec_61850_PIOC">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT"/>
         <Event Name="REQ">
            <With Var="Mod" />
            <With Var="OpCntRs" />
            <With Var="StrVal" />
```



(a) IEC 61499 PIOC FB



(b) IEC 61499 PIOC Sequence Diagram

Figure 7.1: IEC 61499 PIOC FB and Sequence Diagram

```
      </Event>
   </EventInputs>
   <EventOutputs>
      <Event Name="INITO"/>
      <Event Name="CNF"/>
      <Event Name="TRG">
         <With Var="Beh" />
         <With Var="Health" />
         <With Var="Blk" />
         <With Var="Str" />
         <With Var="Op" />
         <With Var="OpCntRs" />
      </Event>
   </EventOutputs>
   <InputVars>
      <VarDeclaration Name="Mod" Type="iec_61850_ENC" />
      <VarDeclaration Name="OpCntRs" Type="iec_61850_INC" />
      <VarDeclaration Name="StrVal" Type="iec_61850_ASG" />
   </InputVars>
   <OutputVars>
      <VarDeclaration Name="Str" Type="iec_61850_ACD" />
      <VarDeclaration Name="Op" Type="iec_61850_ACT" />
      <VarDeclaration Name="NamPlt" Type="iec_61850_LPL" />
      <VarDeclaration Name="Beh" Type="iec_61850_ENS" />
      <VarDeclaration Name="Health" Type="iec_61850_ENS" />
      <VarDeclaration Name="Blk" Type="iec_61850_SPS" />
      <VarDeclaration Name="OpCntRs" Type="iec_61850_INC" />
   </OutputVars>
</InterfaceList>
<Service RightInterface="resource" LeftInterface="iec_61850_PIOC" >
<ServiceSequence Name="setup">
   <ServiceTransaction>
      <InputPrimitive Interface="iec_61850_PIOC" Event="INIT" />
      <OutputPrimitive Interface="resource" Event="init" />
      <OutputPrimitive Interface="iec_61850_PIOC" Event="INITO" />
   </ServiceTransaction>
</ServiceSequence>
<ServiceSequence Name="request">
   <ServiceTransaction>
      <InputPrimitive Interface="iec_61850_PIOC" Event="REQ"
                  Parameters="Mod,OpCntRs,StrVal" />
      <OutputPrimitive Interface="resource" Event="request" />
      <OutputPrimitive Interface="iec_61850_PIOC" Event="CNF" />
   </ServiceTransaction>
```

```
   </ServiceSequence>
   <ServiceSequence Name="trigger" >
      <ServiceTransaction >
         <InputPrimitive Interface="resource" Event="trigger" />
         <OutputPrimitive Interface="iec_61850_PIOC" Event="TRG"
                     Parameters="Str,Op,Beh,Health,Blk,OpCntRs" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.5: Generated IEC 61499 PIOC FB Type

7.3 Used IEC 61850 Elements

7.3.1 Logical Nodes

Common LN				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
Descriptions				
NamPlt	LPL	Name Plate		C
Status information				
Beh	ENS	Behaviour		M
Health	ENS	Health		C
Blk	SPS	Dynamic blocking of function described by the LN		O
Controls				
Mod	ENC	Mode		C
CmdBlk	SPC	Blocking of control sequences and action triggers of controllable data objects		C
Settings				
InRef1	ORG	General input reference		O
BlkRef1	ORG	Blocking reference shows the receiving of dyn. blocking signal		O
Status information				
ClcExp	SPS	Calculation period expired	T	C
Controls				
ClcStr	SPC	Enables the calculation start at time operTm from the control model (if set) or immediately		O
Settings				
ClcMth	ENG	Calculation method of statistical data objects		C
ClcMod	ENG	Calculation mode. Allowed values: TOTAL, PERIOD, SLIDING		C
ClcIntvTyp	ENG	Calculation interval type		C
ClcIntvPer	ING	In case ClcIntvTyp equals to MS, PER-CYCLE, CYCLE, DAY, WEEK, MONTH, YEAR, number of units to consider to calculate the calculation interval duration		C
NumSubIntv	ING	The number of sub-intervals a period interval duration contains		O
ClcRfTyp	ENG	Refreshment interval type		O
ClcRfPer	ING	In case ClcIntvTyp equals to MS, PER-CYCLE, CYCLE, DAY, WEEK, MONTH, YEAR, number of units to consider to calculate the refreshment interval duration		O
ClcSrc	ORG	Object reference to source logical node		C
ClcNxTmms	ING	Remaining time up to the end of the current calculation interval		O
InSyn	ORG	Object ref. to the source of the external synchronization signal		O

Table 7.7: Common LN

CSWI				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Status information				
LocKey	SPS	Local or remote key		O
Loc	SPS	Local control behavior		O
OpOpn	ACT	Operation "Open switch"	T	O
SelOpn	SPS	Selection "Open switch"		O
OpCls	ACT	Operation "Close switch"	T	O
SelCls	SPS	Selection "Close switch"		O
Controls				
OpCntRs	INC	Resettable operation counter		O
LocSta	SPC	Switching authority at station level		O
Pos	DPC	Switch, general		M
PosA	DPC	Switch L1		O
PosB	DPC	Switch L2		O
PosC	DPC	Switch L3		O

Table 7.8: CSWI Logical Node

MMXU				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Measured and metered values				
TotW	MV	Total active power (total P)		O
TotVAr	MV	Total reactive power (total Q)		O
TotVA	MV	Total apparent power (total S)		O
TotPF	MV	Average power factor (total PF)		O
Hz	MV	Frequency		O
PPV	DEL	Phase to phase voltages (VL1,VL2,...)		O
PNV	WYE	Phase to neutral voltage		O
PhV	WYE	Phase to ground voltages (VL1ER,...)		O
A	WYE	Phase currents (IL1, IL2, IL3)		O
W	WYE	Phase active power (P)		O
VAr	WYE	Phase reactive power (Q)		O
VA	WYE	Phase apparent power (S)		O
PF	WYE	Phase power factor		O
Z	WYE	Phase impedance		O
...	...	...		O
Settings				
ClcTotVA	ENG	Calculation method used for total apparent power (TotVA)		O
PFSign	ENG	Sign convention for VAr and power factor (PF)		O

Table 7.9: MMXU Logical Node

PIOC				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Status information				

Str	ACD	Start		O
Op	ACT	Operate		M
Controls				
OpCntRs	INC	Resettable operation counter		O
Settings				
CBOpCap	ASG	Start value		O

Table 7.10: PIOC Logical Node

PTRC				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Status information				
Tr	ACT	Trip		C
Op	ACT	Operate		M
Str	ACD	Start		O
Controls				
OpCntRs	INC	Resettable operation counter		O
Settings				
TrMod	ENG	Trip mode		O
TrPlsTmms	ING	Trip pulse time		O

Table 7.11: PTRC Logical Node

TCTR				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Descriptions				
EEName	DPL	External equipment name plate		O
Status information				
EEHealth	ENS	External equipment health		O
OpTmh	INS	Operation time		M
Measured and metered values				
AmpSv	SAV	Current (sampled value)		O
Settings				
CBOpCap	ASG	Start value		O
ARtg	ASG	Rated current		O
HzRtg	ASG	Rated frequency		O
Rat	ASG	Winding ratio of an external current transformer (transducer) if applicable		O
Cor	ASG	Current phasor magnitude correction of an external current transformer		C
AngCor	ASG	Current phasor angle correction of an external current transformer		C
CorCrv	CSG	Curve phasor magnitude and angle correction		C

Table 7.12: TCTR Logical Node

XCBR				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				

Common Logical Node Information				
		LN shall inherit all Mandatory Data from Common Logical Node Class		M
Desciptions				
EEName	DPL	External equipment name plate		O
Status Information				
Loc	SPS	Local operation (local means without substation automation communication, hardware direct control)		M
EEHealth	INS	External equipment health		O
OpCnt	INS	Operating counter		M
Controls				
Pos	DPC	Switch position		M
BlkOpn	SPC	Block opening		M
BlkCls	SPC	Block closing		M
ChaMotEna	SPC	Charger motor enabled		O
Metered Values				
SumSwARs	BCR	Sum of switched Amperes, resetable		O
Status Information				
CBOpCap	INS	Circuit breaker operation capability		M
POWCap	INS	Point on Wave switching capability		O
MaxOpCap	INS	Circuit breaker operating capability when fully charged		O

Table 7.13: XCBR Logical Node

XSWI				
Attribute Name	**Attribute Type**	**Explanation**	**T**	**M/O**
LNName		Shall be inherited from Logical Node class (see IEC 61850-7-2)		
Data				
Desciptions				
EEName	DPL	External equipment name plate		O
Status Information				
EEHealth	INS	External equipment health		O
LocKey	SPS	Local operation (local means without substation automation communication, hardware direct control)		M
Loc	SPS	Local control behaviour		M
OpCnt	INS	Operating counter		M
CBOpCap	ENS	Circuit breaker operating capability		O
POWCap	ENS	Point on wave switching capability		O
MaxOpCap	INS	Circuit breaker operating capability when fully charged		O
Dsc	SPS	Discrepancy		O
Controls				
LocSta	SPC	Switching authority at station level		O
Pos	DPC	Switch position		M
BlkOpn	SPC	Block opening		M
BlkCls	SPC	Block closing		M
ChaMotEna	SPC	Charger motor enabled		O
Settings				
CBTmms	ING	Closing time of breaker		O

Table 7.14: XSWI Logical Node

7.3.2 Common Data Objects

ACD - Directional protection activation information					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
general	BOOLEAN	ST	dchg		M
dirGeneral	ENUMERATED	ST	dchg	unknown|forward|backward|both	M
physA	BOOLEAN	ST	dchg		O
dirPhsA	ENUMERATED	ST	dchg	unknown|forward|backward	O
physB	BOOLEAN	ST	dchg		O

dirPhsB	ENUMERATED	ST	dchg	unknown|forward|backward	O
physC	BOOLEAN	ST	dchg		O
dirPhsC	ENUMERATED	ST	dchg	unknown|forward|backward	O
neut	BOOLEAN	ST	dchg		O
dirNeut	ENUMERATED	ST	dchg	unknown|forward|backward	O
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
configuration, description and extension					
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.15: ACD Common Data Class

ACT - Protection activation information					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
general	BOOLEAN	ST	dchg		M
physA	BOOLEAN	ST	dchg		O
physB	BOOLEAN	ST	dchg		O
physC	BOOLEAN	ST	dchg		O
neut	BOOLEAN	ST	dchg		O
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
originSrc	Originator	ST			O
operTmPhsA	TimeStamp	ST			O
operTmPhsB	TimeStamp	ST			O
operTmPhsC	TimeStamp	ST			O
configuration, description and extension					
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.16: ACT Common Data Class

ASG - Analogue Setting					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
setting					
setMag	AnalogueValue	SP	dchg		AC_NSG_M
setMag	AnalogueValue	SG, SE			AC_SG_M
configuration, description and extension					
units	Unit	CF	dchg		O
sVC	ScaledValueConfig	CF	dchg		AC_SCAV
minVal	AnalogueValue	CF	dchg		O
maxVal	AnalogueValue	CF	dchg		O
stepSize	AnalogueValue	CF	dchg	1 ... (maxVal - minVal)	O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.17: ASG Common Data Class

BCR - Binary counter reading					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
actVal	INT 64	ST	dchg		M
frVal	INT 64	ST	dupd		GC_2_1
frTm	TimeStamp	ST			GC_2_1
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
configuration, description and extension					
units	Unit	CF	dchg		O
pulsQty	FLOAT 32	CF	dchg		M
frEna	BOOLEAN	CF	dchg		GC_2_1
strTm	TimeStamp	CF	dchg		GC_2_1
frPd	INT 32	CF	dchg		GC_2_1
frRs	BOOLEAN	CF	dchg		GC_2_1
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.18: BCR Common Data Class

CMV - Complex measured value					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
measured attributes					
instCVal	Vector	MX			O
cVal	Vector	MX	dchg, dupd		M
range	ENUMERATED	MX	dchg	normal|high|low|high-high|low-low	O
rangeAng	ENUMERATED	MX	dchg	normal|high|low|high-high|low-low	O
q	Quality	MX	qchg		M
t	TimeStamp	MX			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subCVal	BOOLEAN	SV			PICS_SUBST
subQ	Vector	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
units	Unit	CF	dchg		O
db	INT32U	CF	dchg	0 ... 100000	O
dbAng	INT32U	CF	dchg	0 ... 100000	O
zeroDb	INT32U	CF	dchg	0 ... 100000	O
sVC	ScaledValueConfig	CF	dchg		O
rangeC	RangeConfig	CF	dchg		GC_CON_range
rangeAngC	RangeConfig	CF	dchg		GC_CON_range Ang
magSVC	ScaledValueConfig	CF	dchg		AC_SCAV
angSVC	ScaledValueConfig	CF	dchg		AC_SCAV
angRef	ENUMERATED	CF	dchg	Va|Vb|Vc|Aa|Ab|Ac|Vab|Vbc|Vca|Vother|Aother|Synchrophasor	O
smpRate	INT32U	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.19: CMV Common Data Class

CSG - Curve shape setting					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
settings					
pointZ	FLOAT32	SP			AC_NSG_O
numPts	INT16U	SP		1 < numPts ≤ max-Pts	AC_NSG_M
crvPts	ARRAY 0..maxPts-1	SP			AC_NSG_M
pointZ	FLOAT32	SG, SE			AC_SG_O
numPts	INT16U	SG, SE		1 < numPts ≤ max-Pts	AC_SG_M
crvPts	ARRAY 0..maxPts-1	SG, SE			AC_SG_M
configuration, description and extension					
xUnits	Unit	CF			M
yUnits	Unit	CF			M
zUnits	Unit	CF			O
maxPts	INT16U	CF			M
xD	VISIBLE STRING255	DC			M
xDU	UNICODE STRING255	DC			O
yD	VISIBLE STRING255	DC			M
yDU	UNICODE STRING255	DC			O
zD	VISIBLE STRING255	DC			O
zDU	UNICODE STRING255	DC			O
d	VISIBLE STRING255	DC			O
dU	UNICODE STRING255	DC			O
cdcNs	VISIBLE STRING255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M

Table 7.20: CSG Common Data Class

DEL - Phase to phase related measured values					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Sub Data Objects					
phsAB	CMV				GC_1
phsBC	CMV				GC_1
phsCA	CMV				GC_1
Data Attribute					
configuration, description and extension					
angRef	ENUMERATED	CF	dchg	Va|Vb|Vc|Aa|Ab|Ac|Vab|Vbc|Vca|Vother|Aother|Synchrophasor	O
phsToNeut	BOOLEAN	CF	dchg	DEFAULT=FALSE	O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.21: DEL Common Data Class

DPC - Controllable Double Point					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status and control mirror					
origin	Originator	ST			AC_CO_O
ctlNum	INT8U	ST		0 ... 255	AC_CO_O
stVal	CODED ENUM	ST	dchg	intermediate-state|off|on|bad-state	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
stSeld	BOOLEAN	ST	dchg		O

opRcvd	BOOLEAN	OR	dchg		O
opOk	BOOLEAN	OR	dchg		O
tOpOk	TimeStamp	OR			O
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	CODED ENUM	SV		intermediate-state|off|on|bad-state	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
pulseConfig	PulseConfig	CF	dchg		AC_CO_O
ctlModel	CtlModels	CF	dchg		M
sboTimeout	INT32U	CF	dchg		AC_CO_O
sboClass	SboClasses	CF	dchg		AC_CO_O
operTimeout	INT32U	CF	dchg		AC_CO_O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.22: DPC Common Data Class

DPL - Device Name Plate					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
configuration, description and extension					
vendor	VISIBLE STRING 255	DC			M
hwRev	VISIBLE STRING 255	DC			O
swRev	VISIBLE STRING 255	DC			O
serNum	VISIBLE STRING 255	DC			O
model	VISIBLE STRING 255	DC			O
location	VISIBLE STRING 255	DC			O
name	VISIBLE STRING 64	DC			O
owner	VISIBLE STRING 255	DC			O
ePSName	VISIBLE STRING 255	DC			O
primeOper	VISIBLE STRING 255	DC			O
secondOper	VISIBLE STRING 255	DC			O
latitude	FLOAT 32	DC			O
longitude	FLOAT 32	DC			O
altitude	FLOAT 32	DC			O
mRID	VISIBLE STRING 255	DC			O
d	VISIBLE STRING 255	DC			O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.23: DPL Common Data Class

ENC - Controllable Enumerated Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status and control mirror					
origin	Originator	ST			AC_CO_O
ctlNum	INT8U	ST		0 ... 255	AC_CO_O
stVal	ENUMERATED	ST	dchg	TRUE | FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
stSeld	BOOLEAN	ST	dchg		O
opRcvd	BOOLEAN	OR	dchg		O
opOk	BOOLEAN	OR	dchg		O

tOpOk	TimeStamp	OR			O
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	ENUMERATED	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
ctlModel	CtlModels	CF	dchg		M
sboTimeout	INT32U	CF	dchg		AC_CO_O
sboClass	SboClasses	CF	dchg		AC_CO_O
operTimeout	INT32U	CF	dchg		AC_CO_O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.24: ENC Common Data Class

ENG - Enumerated Status Setting					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
setting					
setVal	ENUMERATED	SP	dchg		AC_NSG_M
setVal	ENUMERATED	SG, SE			AC_SG_M
configuration, description and extension					
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.25: ENG Common Data Class

ENS - Enumerated Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
stVal	ENUMERATED	ST	dchg, dupd		M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	ENUMERATED	SV			PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.26: ENS Common Data Class

INC - Controllable Integer Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**

DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status and control mirror					
origin	Originator	ST			AC_CO_O
ctlNum	INT8U	ST		0 ... 255	AC_CO_O
stVal	INT32	ST	dchg	TRUE | FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
stSeld	BOOLEAN	ST	dchg		O
opRcvd	BOOLEAN	OR	dchg		O
opOk	BOOLEAN	OR	dchg		O
tOpOk	TimeStamp	OR			O
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	INT32	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
ctlModel	CtlModels	CF	dchg		M
sboTimeout	INT32U	CF	dchg		AC_CO_O
sboClass	SboClasses	CF	dchg		AC_CO_O
minVal	INT32	CF	dchg		O
maxVal	INT32	CF	dchg		O
stepSize	INT32U	CF	dchg	1 ... (maxVal - minVal)	O
operTimeout	INT32U	CF	dchg		AC_CO_O
units	Unit	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.27: INC Common Data Class

ING - Integer Status Setting					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
setting					
setVal	INT32	SP	dchg		AC_NSG_M
setVal	INT32	SG, SE			AC_SG_M
configuration, description and extension					
minVal	INT32	CF	dchg		M
maxVal	INT32	CF	dchg		M
stepSize	INT32U	CF	dchq	1 ... (maxVal - minVal)	M
units	Unit	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.28: ING Common Data Class

INS- Integer Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
stVal	INT32	ST	dchg, dupd		M
q	Quality	ST	qchg		M

t	TimeStamp	ST			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	INT32	SV			PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
units	Unit	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.29: INS Common Data Class

LPL - Logical Node Name Plate					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
configuration, description and extension					
vendor	VISIBLE STRING 255	DC			M
swRev	VISIBLE STRING 255	DC			M
d	VISIBLE STRING 255	DC			O
dU	UNICODE STRING 255	DC			O
configRev	VISIBLE STRING 255	DC			AC_LN0_M
paramRev	INT 32	ST	dchg		O
valRev	INT 32	ST	dchg		O
ldNs	VISIBLE STRING 255	EX			AC_LN0_EX
lnNs	VISIBLE STRING 255	EX			AC_DLD_M
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.30: LPL Common Data Class

MV - Measured value					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
measured attributes					
instMag	AnalogueValue	MX			O
mag	AnalogueValue	MX	dchg, dupd		M
range	ENUMERATED	MX	dchg	normal|high|low|high-high|low-low	O
q	Quality	MX	qchg		M
t	TimeStamp	MX			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
units	Unit	CF	dchg		O
db	INT32U	CF	dchg	0 ... 100000	O
zeroDb	INT32U	CF	dchg	0 ... 100000	O
sVC	ScaledValueConfig	CF	dchg		AC_SCAV
rangeC	RangeConfig	CF	dchg		GC_COM_range
smpRate	INT32U	CF	dchg		O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M

cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.31: MV Common Data Class

ORG - Object Reference Setting					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
setting					
setSrcRef	ObjectReference	SP	dchg	Object Reference	M
setTstRef	ObjectReference	SP	dchg	Object Reference	GC_2_1
setSrcCB	ObjectReference	SP	dchg	Object Reference	O
setTstCB	ObjectReference	SP	dchg	Object Reference	GC_CON_set TstRef
intAddr	VISIBLE STRING 255	SP	dchg		O
tstEna	BOOLEAN	SP	dchg		GC_2_1
configuration, description and extension					
purpose	VISIBLE STRING 255	DC			O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.32: ORG Common Data Class

SAV - Sampled value					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
measured attributes					
instMag	AnalogueValue	MX			M
q	Quality	MX	qchg		M
t	TimeStamp	MX			O
configuration, description and extension					
units	Unit	CF	dchg		O
sVC	ScaledValueConfig	CF	dchg		AC_SCAV
min	AnalogueValue	CF	dchg		O
max	AnalogueValue	CF	dchg		O
d	VISIBLE STRING255	DC			O
dU	UNICODE STRING255	DC			O
cdcNs	VISIBLE STRING255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING255	EX			AC_DLN_M

Table 7.33: SAV Common Data Class

SPC - Controllable Single Point					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status and control mirror					
origin	Originator	ST			AC_CO_O
ctlNum	INT8U	ST		0 ... 255	AC_CO_O
stVal	BOOLEAN	ST	dchg	TRUE | FALSE	AC_ST
q	Quality	ST	qchg		AC_ST
t	TimeStamp	ST			AC_ST
stSeld	BOOLEAN	ST	dchg		O
opRcvd	BOOLEAN	OR	dchg		O
opOk	BOOLEAN	OR	dchg		O
tOpOk	TimeStamp	OR			O

substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
pulseConfig	PulseConfig	CF	dchg		AC_CO_O
ctlModel	CtlModels	CF	dchg		M
sboTimeout	INT32U	CF	dchg		AC_CO_O
sboClass	SboClasses	CF	dchg		AC_CO_O
operTimeout	INT32U	CF	dchg		AC_CO_O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.34: SPC Common Data Class

SPS - Single Point Status					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Data Attribute					
status					
stVal	BOOLEAN	ST	dchg, dupt	TRUE | FALSE	M
q	Quality	ST	qchg		M
t	TimeStamp	ST			M
substitution and blocked					
subEna	BOOLEAN	SV			PICS_SUBST
subVal	BOOLEAN	SV		TRUE | FALSE	PICS_SUBST
subQ	Quality	SV			PICS_SUBST
subID	VISIBLE STRING 64	SV			PICS_SUBST
blkEna	BOOLEAN	BL			O
configuration, description and extension					
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.35: SPS Common Data Class

WYE - Phase to ground/neutral related measured values					
Attribute Name	**Attribute Type**	**FC**	**TrgOp**	**Value/Value Range**	**M/O/C**
DataName	Inheritet from Data Class (see 61850-7-2)				
Sub Data Objects					
phsA	CMV				GC_1
phsB	CMV				GC_1
phsC	CMV				GC_1
neut	CMV				GC_1
net	CMV				GC_1
res	CMV				GC_1
Data Attribute					
configuration, description and extension					
angRef	ENUMERATED	CF	dchg	Va|Vb|Vc|Aa|Ab|Ac|Vab|Vbc| Vca|Vother|Aother|Synchrophasor	O
phsToNeut	BOOLEAN	CF	dchg	DEFAULT=FALSE	O
d	VISIBLE STRING 255	DC		Text	O
dU	UNICODE STRING 255	DC			O
cdcNs	VISIBLE STRING 255	EX			AC_DLNDA_M
cdcName	VISIBLE STRING 255	EX			AC_DLNDA_M
dataNs	VISIBLE STRING 255	EX			AC_DLN_M

Table 7.36: WYE Common Data Class

7.4 Generated IEC 61499 Elements

7.4.1 Data Types

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_acd">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="general" Type="iec_61850_boolean"/>
      <VarDeclaration Name="dirGeneral" Type="iec_61850_acd_dir_general"/>
      <VarDeclaration Name="phsA" Type="iec_61850_boolean"/>
      <VarDeclaration Name="dirPhsA" Type="iec_61850_acd_dir"/>
      <VarDeclaration Name="phsB" Type="iec_61850_boolean"/>
      <VarDeclaration Name="dirPhsB" Type="iec_61850_acd_dir"/>
      <VarDeclaration Name="phsC" Type="iec_61850_boolean"/>
      <VarDeclaration Name="dirPhsC" Type="iec_61850_acd_dir"/>
      <VarDeclaration Name="neut" Type="iec_61850_boolean"/>
      <VarDeclaration Name="dirNeut" Type="iec_61850_acd_dir"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.6: Generated IEC 61499 ACD Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_acd_dir">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="unknown"/>
      <EnumeratedValue Name="forward"/>
      <EnumeratedValue Name="backward"/>
   </EnumeratedType>
</DataType>
```

Listing 7.7: Generated IEC 61499 ACD Dir Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_acd_dir_general">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="unknown"/>
      <EnumeratedValue Name="forward"/>
      <EnumeratedValue Name="backward"/>
      <EnumeratedValue Name="both"/>
   </EnumeratedType>
</DataType>
```

Listing 7.8: Generated IEC 61499 ACD Dir General Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_act">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="general" Type="iec_61850_boolean"/>
      <VarDeclaration Name="phsA" Type="iec_61850_boolean"/>
      <VarDeclaration Name="phsB" Type="iec_61850_boolean"/>
      <VarDeclaration Name="phsC" Type="iec_61850_boolean"/>
      <VarDeclaration Name="neut" Type="iec_61850_boolean"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="originSrc" Type="iec_61850_originator"/>
      <VarDeclaration Name="operTmPhsA" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="operTmPhsB" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="operTmPhsC" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.9: Generated IEC 61499 ACT Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_analogue_value">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="i" Type="iec_61850_int32"/>
      <VarDeclaration Name="f" Type="iec_61850_float32"/>
   </StructuredType>
</DataType>
```

Listing 7.10: Generated IEC 61499 Analogue Value Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_asg">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="setMag" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="units" Type="iec_61850_unit"/>
      <VarDeclaration Name="sVC" Type="iec_61850_scaled_value_config"/>
      <VarDeclaration Name="minVal" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="stepSize" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="d" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.11: Generated IEC 61499 ASG Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_boolean">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
```

```
   <DirectlyDerivedType BaseType="BOOL"/>
</DataType>
```

Listing 7.12: Generated IEC 61499 Boolean Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_ctl_models">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="status-only"/>
      <EnumeratedValue Name="direct-with-normal-security"/>
      <EnumeratedValue Name="sbo-with-normal-security"/>
      <EnumeratedValue Name="direct-with-enhanced-security"/>
      <EnumeratedValue Name="sbo-with-enhanced-security"/>
   </EnumeratedType>
</DataType>
```

Listing 7.13: Generated IEC 61499 Ctl Models Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_detail_quality">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="overflow" Type="iec_61850_boolean"/>
      <VarDeclaration Name="outOfRange" Type="iec_61850_boolean"/>
      <VarDeclaration Name="badReference" Type="iec_61850_boolean"/>
      <VarDeclaration Name="oscillatory" Type="iec_61850_boolean"/>
      <VarDeclaration Name="failure" Type="iec_61850_boolean"/>
      <VarDeclaration Name="oldData" Type="iec_61850_boolean"/>
      <VarDeclaration Name="inconsistent" Type="iec_61850_boolean"/>
      <VarDeclaration Name="inaccurate" Type="iec_61850_boolean"/>
   </StructuredType>
</DataType>
```

Listing 7.14: Generated IEC 61499 Detail Quality Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_dpc">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="origin" Type="iec_61850_originator"/>
      <VarDeclaration Name="ctlNum" Type="iec_61850_int8u"/>
      <VarDeclaration Name="stVal" Type="iec_61850_dpc_val"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="stSeld" Type="iec_61850_boolean"/>
      <VarDeclaration Name="opRcvd" Type="iec_61850_boolean"/>
      <VarDeclaration Name="opOk" Type="iec_61850_boolean"/>
      <VarDeclaration Name="tOpOk" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="subEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subVal" Type="iec_61850_dpc_val"/>
      <VarDeclaration Name="subQ" Type="iec_61850_quality"/>
      <VarDeclaration Name="subID" Type="iec_61850_visible_string_64"/>
      <VarDeclaration Name="blkEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="pulseConfig" Type="iec_61850_pulse_config"/>
      <VarDeclaration Name="ctlModel" Type="iec_61850_ctl_models"/>
      <VarDeclaration Name="sboTimeout" Type="iec_61850_int32u"/>
      <VarDeclaration Name="sboClass" Type="iec_61850_sbo_classes"/>
      <VarDeclaration Name="operTimeout" Type="iec_61850_int32u"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
```

```
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.15: Generated IEC 61499 DPC Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_dpc_val">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="intermediate-state"/>
      <EnumeratedValue Name="off"/>
      <EnumeratedValue Name="on"/>
      <EnumeratedValue Name="bad-state"/>
   </EnumeratedType>
</DataType>
```

Listing 7.16: Generated IEC 61499 DPC Val Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_float32">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="REAL"/>
</DataType>
```

Listing 7.17: Generated IEC 61499 Float32 Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_ins">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="stVal" Type="iec_61850_int32"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="subEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subVal" Type="iec_61850_int32"/>
      <VarDeclaration Name="subQ" Type="iec_61850_quality"/>
      <VarDeclaration Name="subID" Type="iec_61850_visible_string_64"/>
      <VarDeclaration Name="blkEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="units" Type="iec_61850_unit"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.18: Generated IEC 61499 INS Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_int8u">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="SINT"/>
```

```
</DataType>
```

Listing 7.19: Generated IEC 61499 Int8u Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_int24u">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <SubrangeType BaseType="UDINT">
      <Subrange LowerLimit="0" UpperLimit="1677215"/>
   </SubrangeType>
</DataType>
```

Listing 7.20: Generated IEC 61499 Int24u Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_int32">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="DINT"/>
</DataType>
```

Listing 7.21: Generated IEC 61499 Int32 Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_int32u">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="UDINT"/>
</DataType>
```

Listing 7.22: Generated IEC 61499 Int32u Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_mv">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="instMag" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="mag" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="range" Type="iec_61850_mv_range"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="subEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subMag" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="subQ" Type="iec_61850_quality"/>
      <VarDeclaration Name="subID" Type="iec_61850_visible_string_64"/>
      <VarDeclaration Name="blkEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="units" Type="iec_61850_unit"/>
      <VarDeclaration Name="db" Type="iec_61850_int32u"/>
      <VarDeclaration Name="zeroDb" Type="iec_61850_int32u"/>
      <VarDeclaration Name="sVC" Type="iec_61850_scaled_value_config"/>
      <VarDeclaration Name="rangeC" Type="iec_61850_range_config"/>
      <VarDeclaration Name="smpRate" Type="iec_61850_int32u"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
```

```
</DataType>
```

Listing 7.23: Generated IEC 61499 MV Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_mv_range">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="normal"/>
      <EnumeratedValue Name="high"/>
      <EnumeratedValue Name="low"/>
      <EnumeratedValue Name="high-high"/>
      <EnumeratedValue Name="low-low"/>
   </EnumeratedType>
</DataType>
```

Listing 7.24: Generated IEC 61499 MV Range Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<!-- Other String Types accordingly -->
<DataType Name="iec_61850_octet_string_64">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="LWORD"/>
</DataType>
```

Listing 7.25: Generated IEC 61499 Octet String 64 Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_originator">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="orCat" Type="iec_61850_originator_orcat"/>
      <VarDeclaration Name="orIdent" Type="iec_61850_octet_string_64"/>
   </StructuredType>
</DataType>
```

Listing 7.26: Generated IEC 61499 Originator Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_originator_orcat">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="not-supported"/>
      <EnumeratedValue Name="bay-control"/>
      <EnumeratedValue Name="station-control"/>
      <EnumeratedValue Name="remote-control"/>
      <EnumeratedValue Name="automatic-bay"/>
      <EnumeratedValue Name="automatic-station"/>
      <EnumeratedValue Name="automatic-remote"/>
      <EnumeratedValue Name="maintenance"/>
      <EnumeratedValue Name="process"/>
   </EnumeratedType>
</DataType>
```

Listing 7.27: Generated IEC 61499 Originator Orcat Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_pulse_config">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="cmdQual" Type="iec_61850_pulse_config_cmdqual"/>
      <VarDeclaration Name="onDur" Type="iec_61850_int32u"/>
      <VarDeclaration Name="offDur" Type="iec_61850_int32u"/>
      <VarDeclaration Name="numPls" Type="iec_61850_int32u"/>
   </StructuredType>
</DataType>
```

Listing 7.28: Generated IEC 61499 Pulse Config Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_pulse_config_cmdqual">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="pulse"/>
      <EnumeratedValue Name="persistent"/>
   </EnumeratedType>
</DataType>
```

Listing 7.29: Generated IEC 61499 Pulse Config Cmdqual Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_quality">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="validity" Type="iec_61850_validity"/>
      <VarDeclaration Name="detailQual" Type="iec_61850_detail_quality"/>
      <VarDeclaration Name="source" Type="iec_61850_source" InitialValue="process"/>
      <VarDeclaration Name="test" Type="iec_61850_boolean" InitialValue="FALSE"/>
      <VarDeclaration Name="operatorBlocked" Type="iec_61850_boolean" InitialValue="FALSE"/>
   </StructuredType>
</DataType>
```

Listing 7.30: Generated IEC 61499 Quality Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_source">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="process"/>
      <EnumeratedValue Name="substituted"/>
   </EnumeratedType>
</DataType>
```

Listing 7.31: Generated IEC 61499 Quality Source Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_range_config">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
```

```
      <VarDeclaration Name="hhLim" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="hLim" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="ILim" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="IILim" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="min" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="max" Type="iec_61850_analogue_value"/>
      <VarDeclaration Name="limDb" Type="iec_61850_int32u"/>
   </StructuredType>
</DataType>
```

Listing 7.32: Generated IEC 61499 Range Config Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_validity">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="operate-once"/>
      <EnumeratedValue Name="operate-many"/>
   </EnumeratedType>
</DataType>
```

Listing 7.33: Generated IEC 61499 SBO Classes Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_scaled_value_config">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="scaleFactor" Type="iec_61850_float32"/>
      <VarDeclaration Name="offset" Type="iec_61850_float32"/>
   </StructuredType>
</DataType>
```

Listing 7.34: Generated IEC 61499 Scaled Value Config Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_source">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="process"/>
      <EnumeratedValue Name="substituted"/>
   </EnumeratedType>
</DataType>
```

Listing 7.35: Generated IEC 61499 Source Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_spc">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="origin" Type="iec_61850_originator"/>
      <VarDeclaration Name="ctlNum" Type="iec_61850_int8u"/>
      <VarDeclaration Name="stVal" Type="iec_61850_boolean"/>
      <VarDeclaration Name="q" Type="iec_61850_quality"/>
      <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="stSeld" Type="iec_61850_boolean"/>
      <VarDeclaration Name="opRcvd" Type="iec_61850_boolean"/>
```

```
      <VarDeclaration Name="opOk" Type="iec_61850_boolean"/>
      <VarDeclaration Name="tOpOk" Type="iec_61850_timestamp"/>
      <VarDeclaration Name="subEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subVal" Type="iec_61850_boolean"/>
      <VarDeclaration Name="subQ" Type="iec_61850_quality"/>
      <VarDeclaration Name="subID" Type="iec_61850_visible_string_64"/>
      <VarDeclaration Name="blkEna" Type="iec_61850_boolean"/>
      <VarDeclaration Name="pulseConfig" Type="iec_61850_pulse_config"/>
      <VarDeclaration Name="ctlModel" Type="iec_61850_ctl_models"/>
      <VarDeclaration Name="sboTimeout" Type="iec_61850_int32u"/>
      <VarDeclaration Name="sboClass" Type="iec_61850_sbo_classes"/>
      <VarDeclaration Name="operTimeout" Type="iec_61850_int32u"/>
      <VarDeclaration Name="d" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dU" Type="iec_61850_unicode_string_255"/>
      <VarDeclaration Name="cdcNs" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="cdcName" Type="iec_61850_visible_string_255"/>
      <VarDeclaration Name="dataNs" Type="iec_61850_visible_string_255"/>
   </StructuredType>
</DataType>
```

Listing 7.36: Generated IEC 61499 SPC Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_timeaccuracy">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="7"/>
      <EnumeratedValue Name="10"/>
      <EnumeratedValue Name="14"/>
      <EnumeratedValue Name="16"/>
      <EnumeratedValue Name="18"/>
      <EnumeratedValue Name="20"/>
      <EnumeratedValue Name="31"/>
   </EnumeratedType>
</DataType>
```

Listing 7.37: Generated IEC 61499 Timeaccuracy Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_timequality">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="LeapSecondsKnown" Type="iec_61850_boolean"/>
      <VarDeclaration Name="ClockFailure" Type="iec_61850_boolean"/>
      <VarDeclaration Name="ClockNotSynchronized" Type="iec_61850_boolean"/>
      <VarDeclaration Name="TimeAccuracy" Type="iec_61850_timeaccuracy"/>
   </StructuredType>
</DataType>
```

Listing 7.38: Generated IEC 61499 Timequality Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_timestamp">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="SecondSinceEpoch" Type="iec_61850_int32u"/>
      <VarDeclaration Name="FractionOfSecond" Type="iec_61850_int24u"/>
      <VarDeclaration Name="TimeQuality" Type="iec_61850_timequality"/>
   </StructuredType>
</DataType>
```

Listing 7.39: Generated IEC 61499 Timestamp Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<!-- Other String Types accordingly -->
<DataType Name="iec_61850_unicode_string_255">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="STRING(255)"/>
</DataType>
```

Listing 7.40: Generated IEC 61499 Unicode String 255 Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_unit">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <StructuredType>
      <VarDeclaration Name="SIUnit" Type="iec_61850_unit_siunit"/>
      <VarDeclaration Name="multiplier" Type="iec_61850_unit_multiplier"/>
   </StructuredType>
</DataType>
```

Listing 7.41: Generated IEC 61499 Unit Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_unit_multiplier">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="-24"/>
      <EnumeratedValue Name="-21"/>
      <EnumeratedValue Name="-18"/>
      <EnumeratedValue Name="-15"/>
      <EnumeratedValue Name="-12"/>
      <EnumeratedValue Name="-9"/>
      <EnumeratedValue Name="-6"/>
      <EnumeratedValue Name="-3"/>
      <EnumeratedValue Name="-2"/>
      <EnumeratedValue Name="-1"/>
      <EnumeratedValue Name="0"/>
      <EnumeratedValue Name="1"/>
      <EnumeratedValue Name="2"/>
      <EnumeratedValue Name="3"/>
      <EnumeratedValue Name="6"/>
      <EnumeratedValue Name="9"/>
      <EnumeratedValue Name="12"/>
      <EnumeratedValue Name="15"/>
      <EnumeratedValue Name="18"/>
      <EnumeratedValue Name="21"/>
      <EnumeratedValue Name="24"/>
   </EnumeratedType>
</DataType>
```

Listing 7.42: Generated IEC 61499 Unit Multiplier Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_unit_siunit">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="1"/>
      <EnumeratedValue Name="2"/>
      <EnumeratedValue Name="3"/>
      <EnumeratedValue Name="4"/>
```

```
<EnumeratedValue Name="5"/>
<EnumeratedValue Name="6"/>
<EnumeratedValue Name="7"/>
<EnumeratedValue Name="8"/>
<EnumeratedValue Name="9"/>
<EnumeratedValue Name="10"/>
<EnumeratedValue Name="11"/>
<EnumeratedValue Name="12"/>
<EnumeratedValue Name="13"/>
<EnumeratedValue Name="14"/>
<EnumeratedValue Name="15"/>
<EnumeratedValue Name="16"/>
<EnumeratedValue Name="17"/>
<EnumeratedValue Name="18"/>
<EnumeratedValue Name="19"/>
<EnumeratedValue Name="20"/>
<EnumeratedValue Name="21"/>
<EnumeratedValue Name="22"/>
<EnumeratedValue Name="23"/>
<EnumeratedValue Name="24"/>
<EnumeratedValue Name="25"/>
<EnumeratedValue Name="26"/>
<EnumeratedValue Name="27"/>
<EnumeratedValue Name="28"/>
<EnumeratedValue Name="29"/>
<EnumeratedValue Name="30"/>
<EnumeratedValue Name="31"/>
<EnumeratedValue Name="32"/>
<EnumeratedValue Name="33"/>
<EnumeratedValue Name="34"/>
<EnumeratedValue Name="35"/>
<EnumeratedValue Name="36"/>
<EnumeratedValue Name="37"/>
<EnumeratedValue Name="38"/>
<EnumeratedValue Name="39"/>
<EnumeratedValue Name="40"/>
<EnumeratedValue Name="41"/>
<EnumeratedValue Name="42"/>
<EnumeratedValue Name="43"/>
<EnumeratedValue Name="44"/>
<EnumeratedValue Name="45"/>
<EnumeratedValue Name="46"/>
<EnumeratedValue Name="47"/>
<EnumeratedValue Name="48"/>
<EnumeratedValue Name="49"/>
<EnumeratedValue Name="50"/>
<EnumeratedValue Name="51"/>
<EnumeratedValue Name="52"/>
<EnumeratedValue Name="53"/>
<EnumeratedValue Name="54"/>
<EnumeratedValue Name="55"/>
<EnumeratedValue Name="56"/>
<EnumeratedValue Name="57"/>
<EnumeratedValue Name="58"/>
<EnumeratedValue Name="59"/>
<EnumeratedValue Name="60"/>
<EnumeratedValue Name="61"/>
<EnumeratedValue Name="62"/>
<EnumeratedValue Name="63"/>
<EnumeratedValue Name="64"/>
<EnumeratedValue Name="65"/>
<EnumeratedValue Name="66"/>
<EnumeratedValue Name="67"/>
<EnumeratedValue Name="68"/>
<EnumeratedValue Name="69"/>
<EnumeratedValue Name="70"/>
<EnumeratedValue Name="71"/>
<EnumeratedValue Name="72"/>
<EnumeratedValue Name="73"/>
<EnumeratedValue Name="74"/>
<EnumeratedValue Name="75"/>
<EnumeratedValue Name="76"/>
<EnumeratedValue Name="77"/>
```

```
      <EnumeratedValue Name="78"/>
      <EnumeratedValue Name="79"/>
      <EnumeratedValue Name="80"/>
      <EnumeratedValue Name="81"/>
      <EnumeratedValue Name="82"/>
      <EnumeratedValue Name="83"/>
   </EnumeratedType>
</DataType>
```

Listing 7.43: Generated IEC 61499 Unit SI Unit Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<DataType Name="iec_61850_validity">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <EnumeratedType>
      <EnumeratedValue Name="good"/>
      <EnumeratedValue Name="invalid"/>
      <EnumeratedValue Name="reserved"/>
      <EnumeratedValue Name="questionable"/>
   </EnumeratedType>
</DataType>
```

Listing 7.44: Generated IEC 61499 Validity Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<!-- Other String Types accordingly -->
<DataType Name="iec_61850_visible_string_64">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="STRING(64)"/>
</DataType>
```

Listing 7.45: Generated IEC 61499 Visible String 64 Data Type

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd" >

<!-- Other String Types accordingly -->
<DataType Name="iec_61850_visible_string_255">
   <VersionInfo Organization="TU Wien" Version="1.0" Author="FK" Date="2021-01-01"/>
   <ASN1Tag Class="APPLICATION" Number="1" />
   <DirectlyDerivedType BaseType="STRING(255)"/>
</DataType>
```

Listing 7.46: Generated IEC 61499 Visible String 255 Data Type

7.4.2 Devices, Function Blocks, Resources and Segments

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DeviceType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd" >
<DeviceType Name="FORTE_PC" Comment="FORTE running on a PC (Windows or Linux)" >
  <Identification Standard="61499 ITA" Description="Copyright (c) 2015 fortiss GmbH&#13;&#10;
      &#13;&#10;This program and the accompanying materials are made&#13;&#10;available under the
      terms of the Eclipse Public License 2.0&#13;&#10;which is available at https://www.eclipse.
      org/legal/epl-2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" />
  <VersionInfo Organization="fortiss GmbH" Version="1.0" Author="Alois Zoitl" Date="2015-10-31"
      Remarks="Initial version" />
  <VarDeclaration Name="MGR_ID" Type="WSTRING" InitialValue="&#34;localhost:61499&#34;" Comment="
      Device manager socket ID" />
</DeviceType>
```

Listing 7.47: IEC 61499 FORTE_PC Device

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="COMPOSE_ASG">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="REQ">
            <With Var="setMag" />
            <With Var="units" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="CNF">
            <With Var="asgVal"/>
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="setMag" Type="iec_61850_analogue_value"/>
         <VarDeclaration Name="units" Type="iec_61850_unit"/>
      </InputVars>
      <OutputVars>
         <VarDeclaration Name="asgVal" Type="iec_61850_asg"/>
      </OutputVars>
   </InterfaceList>
   <BasicFB>
      <ECC >
         <ECState Name="START"/>
         <ECState Name="REQ">
            <ECAction Algorithm="REQ" Output="CNF" />
         </ECState>
         <ECTransition Source="START" Destination="REQ" Condition="REQ" />
         <ECTransition Source="REQ" Destination="START" Condition="1" />
      </ECC>
      <Algorithm Name="REQ" >
         <ST Text="asgVal.setMag := setMag; asgVal.units := units;" />
      </Algorithm>
   </BasicFB>
</FBType>
```

Listing 7.48: IEC 61499 COMPOSE_ASG Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="DECOMPOSE_DPC">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="REQ">
            <With Var="dpcVal" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="CNF">
            <With Var="stVal"/>
            <With Var="q"/>
            <With Var="t"/>
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="dpcVal" Type="iec_61850_dpc"/>
      </InputVars>
      <OutputVars>
         <VarDeclaration Name="stVal" Type="iec_61850_dpc_val"/>
         <VarDeclaration Name="q" Type="iec_61850_quality"/>
         <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
      </OutputVars>
   </InterfaceList>
   <BasicFB>
      <ECC >
         <ECState Name="START"/>
         <ECState Name="REQ">
            <ECAction Algorithm="REQ" Output="CNF" />
```

```
        </ECState>
        <ECTransition Source="START" Destination="REQ" Condition="REQ" />
        <ECTransition Source="REQ" Destination="START" Condition="1" />
      </ECC>
      <Algorithm Name="REQ" >
        <ST Text="stVal := dpcVal.stVal; q := dpcVal.q; t := dpcVal.t;" />
      </Algorithm>
   </BasicFB>
</FBType>
```

Listing 7.49: IEC 61499 DECOMPOSE_DPC Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="Peroidic event generator" Name="E_CYCLE">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
       the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
       License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
       -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" Standard="61499-1 Annex A"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-09-21" Organization="fortiss GmbH" Remarks="initial
       API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="Start the perodic generation of events" Name="START" Type="Event">
        <With Var="DT"/>
      </Event>
      <Event Comment="Stop the generation of events" Name="STOP" Type="Event"/>
    </EventInputs>
    <EventOutputs>
      <Event Comment="Periodically triggered output event" Name="EO" Type="Event"/>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Comment="cycle time" Name="DT" Type="TIME"/>
    </InputVars>
    <OutputVars/>
  </InterfaceList>
  <FBNetwork>
    <FB Comment="" Name="E_DELAY" Type="E_DELAY" x="1200.0" y="-100.0"/>
    <DataConnections>
      <Connection Comment="" Destination="E_DELAY.DT" Source="DT" dx1="0.0" dx2="0.0" dy="0.0"/>
    </DataConnections>
    <EventConnections>
      <Connection Comment="" Destination="E_DELAY.START" Source="START" dx1="270.0" dx2="0.0" dy="0.0
           "/>
      <Connection Comment="" Destination="E_DELAY.STOP" Source="STOP" dx1="0.0" dx2="0.0" dy="0.0"/>
      <Connection Comment="" Destination="E_DELAY.START" Source="E_DELAY.EO" dx1="0.0" dx2="0.0" dy="
           -170.0"/>
      <Connection Comment="" Destination="EO" Source="E_DELAY.EO" dx1="0.0" dx2="0.0" dy="0.0"/>
    </EventConnections>
  </FBNetwork>
</FBType>
```

Listing 7.50: IEC 61499 E_CYCLE Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="Delayed event propagation" Name="E_DELAY">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10;&#13;&#10;This program and
       the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
       License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
       -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" Standard="61499-1 Annex A"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-09-21" Organization="fortiss GmbH" Remarks="initial
       API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="Start delayed event propagation" Name="START" Type="Event">
        <With Var="DT"/>
      </Event>
      <Event Comment="Stop the delayed event propagation" Name="STOP" Type="Event"/>
    </EventInputs>
    <EventOutputs>
```

```
    <Event Comment="Delayed event" Name="EO" Type="Event"/>
   </EventOutputs>
   <InputVars>
    <VarDeclaration Comment="Delay time, &gt;0" Name="DT" Type="TIME"/>
   </InputVars>
   <OutputVars/>
  </InterfaceList>
  <Service Comment="Delayed event propagation" LeftInterface="APPLICATION" RightInterface="RESOURCE"
      />
</FBType>
```

Listing 7.51: IEC 61499 E_DELAY Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="Service Interface Function Block Type" Name="E_RESTART">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
      the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
      License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
      -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" Standard="61499-1 Annex A"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-09-21" Organization="fortiss GmbH" Remarks="initial
      API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
   <EventInputs/>
   <EventOutputs>
    <Event Comment="Information on cold restart" Name="COLD" Type="Event"/>
    <Event Comment="Information on warm restart" Name="WARM" Type="Event"/>
    <Event Comment="information that the resource is to be stopped" Name="STOP" Type="Event"/>
   </EventOutputs>
   <InputVars/>
   <OutputVars/>
  </InterfaceList>
  <Service Comment="Service Interface Function Block Type" LeftInterface="E_RESTART" RightInterface=
      "RESOURCE">
   <ServiceSequence Comment="" Name="cold_restart">
    <ServiceTransaction>
     <InputPrimitive Event="start" Interface="RESOURCE"/>
     <OutputPrimitive Event="START" Interface="E_RESTART"/>
    </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Comment="" Name="warm_restart">
    <ServiceTransaction>
     <InputPrimitive Event="restart" Interface="RESOURCE"/>
     <OutputPrimitive Event="WARM" Interface="E_RESTART"/>
    </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Comment="" Name="stopping">
    <ServiceTransaction>
     <InputPrimitive Event="stop" Interface="RESOURCE"/>
     <OutputPrimitive Event="STOP" Interface="E_RESTART"/>
    </ServiceTransaction>
   </ServiceSequence>
  </Service>
</FBType>
```

Listing 7.52: IEC 61499 E_RESTART Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="iec_61850_MMXU">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <Identification Standard="IEC 61850"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT"/>
         <Event Name="REQ"/>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
         <Event Name="TRG">
```

```
            <With Var="TotW" />
            <With Var="TotVAr" />
            <With Var="TotVA" />
            <With Var="TotPF" />
            <With Var="Hz" />
            <With Var="MaxAPhs" />
         </Event>
      </EventOutputs>
      <InputVars>
      </InputVars>
      <OutputVars>
         <VarDeclaration Name="TotW" Type="iec_61850_mv" />
         <VarDeclaration Name="TotVAr" Type="iec_61850_mv" />
         <VarDeclaration Name="TotVA" Type="iec_61850_mv" />
         <VarDeclaration Name="TotPF" Type="iec_61850_mv" />
         <VarDeclaration Name="Hz" Type="iec_61850_mv" />
         <VarDeclaration Name="MaxAPhs" Type="iec_61850_mv" />
      </OutputVars>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="iec_61850_MMXU" >
     <ServiceSequence Name="setup">
        <ServiceTransaction>
           <InputPrimitive Interface="iec_61850_MMXU" Event="INIT" />
           <OutputPrimitive Interface="resource" Event="init" />
           <OutputPrimitive Interface="iec_61850_MMXU" Event="INITO" />
        </ServiceTransaction>
     </ServiceSequence>
     <ServiceSequence Name="request">
        <ServiceTransaction>
           <InputPrimitive Interface="iec_61850_MMXU" Event="REQ"/>
           <OutputPrimitive Interface="resource" Event="request" />
           <OutputPrimitive Interface="iec_61850_MMXU" Event="CNF" />
        </ServiceTransaction>
     </ServiceSequence>
     <ServiceSequence Name="trigger" >
        <ServiceTransaction >
           <InputPrimitive Interface="resource" Event="trigger" />
           <OutputPrimitive Interface="iec_61850_MMXU" Event="TRG"
                       Parameters="TotW,TotVAr,TotVA,TotPF,Hz,MaxAPhs" />
        </ServiceTransaction>
     </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.53: IEC 61499 MMXU Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >

<FBType Name="iec_61850_PIOC">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <Identification Standard="IEC 61850"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT"/>
         <Event Name="REQ">
            <With Var="StrVal" />
            <With Var="instMag" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
         <Event Name="TRG">
            <With Var="Str" />
            <With Var="Op" />
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="asgVal" Type="iec_61850_asg" />
         <VarDeclaration Name="mvVal" Type="iec_61850_mv" />
      </InputVars>
```

```
      <OutputVars>
         <VarDeclaration Name="Op" Type="iec_61850_act" />
         <VarDeclaration Name="Str" Type="iec_61850_acd" />
      </OutputVars>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="iec_61850_PIOC" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_PIOC" Event="INIT" />
         <OutputPrimitive Interface="resource" Event="init" />
         <OutputPrimitive Interface="iec_61850_PIOC" Event="INITO" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_PIOC" Event="REQ"
                     Parameters="StrVal,instMag" />
         <OutputPrimitive Interface="resource" Event="request" />
         <OutputPrimitive Interface="iec_61850_PIOC" Event="CNF" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="trigger" >
      <ServiceTransaction >
         <InputPrimitive Interface="resource" Event="trigger" />
         <OutputPrimitive Interface="iec_61850_PIOC" Event="TRG"
                      Parameters="Str,Op" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.54: IEC 61499 PIOC Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="iec_61850_PTRC">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <Identification Standard="IEC 61850"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT"/>
         <Event Name="REQ">
            <With Var="Op_In" />
            <With Var="Str_In" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
         <Event Name="TRG">
            <With Var="Tr" />
            <With Var="Op_Out" />
            <With Var="Str_Out" />
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="Op_In" Type="iec_61850_act" />
         <VarDeclaration Name="Str_In" Type="iec_61850_acd" />
      </InputVars>
      <OutputVars>
        <VarDeclaration Name="Tr" Type="iec_61850_act" />
        <VarDeclaration Name="Op_Out" Type="iec_61850_act" />
        <VarDeclaration Name="Str_Out" Type="iec_61850_acd" />
      </OutputVars>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="iec_61850_PTRC" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_PTRC" Event="INIT" />
         <OutputPrimitive Interface="resource" Event="init" />
         <OutputPrimitive Interface="iec_61850_PTRC" Event="INITO" />
      </ServiceTransaction>
```

```
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_PTRC" Event="REQ"
                         Parameters="Op_In,Str_In" />
         <OutputPrimitive Interface="resource" Event="request" />
         <OutputPrimitive Interface="iec_61850_PTRC" Event="CNF" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="trigger" >
      <ServiceTransaction >
         <InputPrimitive Interface="resource" Event="trigger" />
         <OutputPrimitive Interface="iec_61850_PTRC" Event="TRG"
                          Parameters="Str_Out,Op_Out,Tr" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.55: IEC 61499 PTRC Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="iec_61850_XCBR">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <Identification Standard="IEC 61850"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT"/>
         <Event Name="REQ">
            <With Var="Tr" />
            <With Var="Pos_In" />
            <With Var="BlkOpn_In" />
            <With Var="BlkCls_In" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
         <Event Name="TRG">
            <With Var="Loc" />
            <With Var="OpCnt" />
            <With Var="Pos_Out" />
            <With Var="BlkOpn_Out" />
            <With Var="BlkCls_Out" />
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="Tr" Type="iec_61850_act" />
         <VarDeclaration Name="Pos_In" Type="iec_61850_dpc" />
         <VarDeclaration Name="BlkOpn_In" Type="iec_61850_spc" />
         <VarDeclaration Name="BlkCls_In" Type="iec_61850_spc" />
      </InputVars>
      <OutputVars>
        <VarDeclaration Name="Loc" Type="iec_61850_sps" />
        <VarDeclaration Name="OpCnt" Type="iec_61850_ins" />
        <VarDeclaration Name="Pos_Out" Type="iec_61850_dpc" />
        <VarDeclaration Name="BlkOpn_Out" Type="iec_61850_spc" />
        <VarDeclaration Name="BlkCls_Out" Type="iec_61850_spc" />
      </OutputVars>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="iec_61850_XCBR" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_XCBR" Event="INIT" />
         <OutputPrimitive Interface="resource" Event="init" />
         <OutputPrimitive Interface="iec_61850_XCBR" Event="INITO" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="iec_61850_XCBR" Event="REQ"
```

```
                    Parameters="Tr,Pos_In,BlkOpn_In,BlkCls_In" />
            <OutputPrimitive Interface="resource" Event="request" />
            <OutputPrimitive Interface="iec_61850_XCBR" Event="CNF" />
        </ServiceTransaction>
    </ServiceSequence>
    <ServiceSequence Name="trigger" >
        <ServiceTransaction >
            <InputPrimitive Interface="resource" Event="trigger" />
            <OutputPrimitive Interface="iec_61850_XCBR" Event="TRG"
                    Parameters="Loc,OpCnt,Pos_Out,BlkOpn_Out.BlkCls_Out" />
        </ServiceTransaction>
    </ServiceSequence>
    </Service>
</FBType>
```

Listing 7.56: IEC 61499 XCBR Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="Pub_GOOSE_iec_61850_asg">
  <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="" Name="INIT" Type="Event">
        <With Var="Ip"/>
        <With Var="ID"/>
      </Event>
      <Event Comment="" Name="REQ" Type="Event">
        <With Var="SD_1"/>
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Comment="" Name="INITO" Type="Event">
      </Event>
      <Event Comment="" Name="CNF" Type="Event">
       </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Comment="" Name="Ip" Type="WSTRING"/>
      <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
      <VarDeclaration Comment="" Name="SD_1" Type="iec_61850_asg"/>
    </InputVars>
    <OutputVars>

    </OutputVars>
  </InterfaceList>
</FBType>
```

Listing 7.57: IEC 61499 PUB_GOOSE_ASG Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="PUBLISH_1">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
      the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
      License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
      -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" Standard="61499-2"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-10-25" Organization="fortiss GmbH" Remarks="initial
      API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="" Name="INIT" Type="Event">
        <With Var="QI"/>
        <With Var="ID"/>
      </Event>
      <Event Comment="" Name="REQ" Type="Event">
        <With Var="QI"/>
        <With Var="SD_1"/>
      </Event>
    </EventInputs>
    <EventOutputs>
```

```
    <Event Comment="" Name="INITO" Type="Event">
      <With Var="QO"/>
      <With Var="STATUS"/>
    </Event>
    <Event Comment="" Name="CNF" Type="Event">
      <With Var="QO"/>
      <With Var="STATUS"/>
    </Event>
   </EventOutputs>
   <InputVars>
    <VarDeclaration Comment="" Name="QI" Type="BOOL"/>
    <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
    <VarDeclaration Comment="" Name="SD_1" Type="ANY"/>
   </InputVars>
   <OutputVars>
    <VarDeclaration Comment="" Name="QO" Type="BOOL"/>
    <VarDeclaration Comment="" Name="STATUS" Type="WSTRING"/>
   </OutputVars>
  </InterfaceList>
</FBType>
```

Listing 7.58: IEC 61499 PUBLISH_1 Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="PUBLISH_3">
 <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
     the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
     License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
     -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" Standard="61499-2"/>
 <VersionInfo Author="Alois Zoitl" Date="2017-10-25" Organization="fortiss GmbH" Remarks="initial
     API and implementation and/or initial documentation" Version="1.0"/>
 <InterfaceList>
  <EventInputs>
   <Event Comment="" Name="INIT" Type="Event">
    <With Var="QI"/>
    <With Var="ID"/>
   </Event>
   <Event Comment="" Name="REQ" Type="Event">
    <With Var="QI"/>
    <With Var="SD_1"/>
    <With Var="SD_2"/>
    <With Var="SD_3"/>
   </Event>
  </EventInputs>
  <EventOutputs>
   <Event Comment="" Name="INITO" Type="Event">
    <With Var="QO"/>
    <With Var="STATUS"/>
   </Event>
   <Event Comment="" Name="CNF" Type="Event">
    <With Var="QO"/>
    <With Var="STATUS"/>
   </Event>
  </EventOutputs>
  <InputVars>
   <VarDeclaration Comment="" Name="QI" Type="BOOL"/>
   <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
   <VarDeclaration Comment="" Name="SD_1" Type="ANY"/>
   <VarDeclaration Comment="" Name="SD_2" Type="ANY"/>
   <VarDeclaration Comment="" Name="SD_3" Type="ANY"/>
  </InputVars>
  <OutputVars>
   <VarDeclaration Comment="" Name="QO" Type="BOOL"/>
   <VarDeclaration Comment="" Name="STATUS" Type="WSTRING"/>
  </OutputVars>
 </InterfaceList>
</FBType>
```

Listing 7.59: IEC 61499 PUBLISH_3 Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="READ_FROM_DB">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT">
            <With Var="ID" />
         </Event>
         <Event Name="REQ"/>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF">
            <With Var="Str" />
         </Event>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="ID" Type="WSTRING" />
      </InputVars>
      <OutputVars>
         <VarDeclaration Name="Str" Type="WSTRING" />
      </OutputVars>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="READ_FROM_DB" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="READ_FROM_DB" Event="INIT" Parameters="ID" />
         <OutputPrimitive Interface="resource" Event="init" />
         <OutputPrimitive Interface="READ_FROM_DB" Event="INITO" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="READ_FROM_DB" Event="REQ" />
         <OutputPrimitive Interface="resource" Event="request" />
         <OutputPrimitive Interface="READ_FROM_DB" Event="CNF" Parameters="Str" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.60: IEC 61499 READ_FROM_DB Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="SEND_SMS">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT">
            <With Var="ID" />
         </Event>
         <Event Name="REQ">
            <With Var="Str" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="Str" Type="WSTRING" />
         <VarDeclaration Name="ID" Type="WSTRING" />
      </InputVars>
      <OutputVars/>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="SEND_SMS" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="SEND_SMS" Event="INIT" Parameters="ID" />
         <OutputPrimitive Interface="resource" Event="init" />
```

```
        <OutputPrimitive Interface="SEND_SMS" Event="INITO" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="SEND_SMS" Event="REQ" Parameters="Str" />
         <OutputPrimitive Interface="resource" Event="request" />
         <OutputPrimitive Interface="SEND_SMS" Event="CNF" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.61: IEC 61499 SEND_SMS Function Block

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "LibraryElement.dtd" >
<FBType Name="STORE_IN_DB">
   <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
   <InterfaceList>
      <EventInputs>
         <Event Name="INIT">
            <With Var="ID" />
         </Event>
         <Event Name="REQ">
            <With Var="stVal" />
            <With Var="q" />
            <With Var="t" />
         </Event>
      </EventInputs>
      <EventOutputs>
         <Event Name="INITO"/>
         <Event Name="CNF"/>
      </EventOutputs>
      <InputVars>
         <VarDeclaration Name="stVal" Type="iec_61850_dpc_val"/>
         <VarDeclaration Name="q" Type="iec_61850_quality"/>
         <VarDeclaration Name="t" Type="iec_61850_timestamp"/>
         <VarDeclaration Name="ID" Type="WSTRING" />
      </InputVars>
      <OutputVars/>
   </InterfaceList>
   <Service RightInterface="resource" LeftInterface="STORE_IN_DB" >
   <ServiceSequence Name="setup">
      <ServiceTransaction>
         <InputPrimitive Interface="STORE_IN_DB" Event="INIT" Parameters="ID" />
         <OutputPrimitive Interface="resource" Event="init" />
         <OutputPrimitive Interface="STORE_IN_DB" Event="INITO" />
      </ServiceTransaction>
   </ServiceSequence>
   <ServiceSequence Name="request">
      <ServiceTransaction>
         <InputPrimitive Interface="STORE_IN_DB" Event="REQ" Parameters="stVal,q,t" />
         <OutputPrimitive Interface="resource" Event="request" />
         <OutputPrimitive Interface="STORE_IN_DB" Event="CNF" />
      </ServiceTransaction>
   </ServiceSequence>
   </Service>
</FBType>
```

Listing 7.62: IEC 61499 STORE_IN_DB Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="Sub_GOOSE_iec_61850_dpc">
  <VersionInfo Organization="TU Wien" Version="1" Author="FK" Date="2021-01-01"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="" Name="INIT" Type="Event">
        <With Var="Ip"/>
        <With Var="Mac"/>
```

```
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Comment="" Name="INITO" Type="Event">
      </Event>
      <Event Comment="" Name="IND" Type="Event">
        <With Var="RD_1"/>
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
      <VarDeclaration Comment="" Name="Mac" Type="WSTRING"/>
    </InputVars>
    <OutputVars>
      <VarDeclaration Comment="" Name="RD_1" Type="iec_61850_dpc"/>
    </OutputVars>
  </InterfaceList>
</FBType>
```

Listing 7.63: IEC 61499 SUB_GOOSE_DPC Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="SUBSCRIBE_1">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
      the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
      License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
      -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-10-25" Organization="fortiss GmbH" Remarks="initial
      API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="" Name="INIT" Type="Event">
        <With Var="QI"/>
        <With Var="ID"/>
      </Event>
      <Event Comment="" Name="RSP" Type="Event">
        <With Var="QI"/>
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Comment="" Name="INITO" Type="Event">
        <With Var="QO"/>
        <With Var="STATUS"/>
      </Event>
      <Event Comment="" Name="IND" Type="Event">
        <With Var="QO"/>
        <With Var="STATUS"/>
        <With Var="RD_1"/>
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Comment="" Name="QI" Type="BOOL"/>
      <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
    </InputVars>
    <OutputVars>
      <VarDeclaration Comment="" Name="QO" Type="BOOL"/>
      <VarDeclaration Comment="" Name="STATUS" Type="WSTRING"/>
      <VarDeclaration Comment="" Name="RD_1" Type="ANY"/>
    </OutputVars>
  </InterfaceList>
</FBType>
```

Listing 7.64: IEC 61499 SUBSCRIBE_1 Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<FBType Comment="" Name="SUBSCRIBE_3">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
      the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
      License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
      -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0"/>
```

```
  <VersionInfo Author="Alois Zoitl" Date="2017-10-25" Organization="fortiss GmbH" Remarks="initial
       API and implementation and/or initial documentation" Version="1.0"/>
  <InterfaceList>
    <EventInputs>
      <Event Comment="" Name="INIT" Type="Event">
        <With Var="QI"/>
        <With Var="ID"/>
      </Event>
      <Event Comment="" Name="RSP" Type="Event">
        <With Var="QI"/>
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Comment="" Name="INITO" Type="Event">
        <With Var="QO"/>
        <With Var="STATUS"/>
      </Event>
      <Event Comment="" Name="IND" Type="Event">
        <With Var="QO"/>
        <With Var="STATUS"/>
        <With Var="RD_1"/>
        <With Var="RD_3"/>
        <With Var="RD_2"/>
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Comment="" Name="QI" Type="BOOL"/>
      <VarDeclaration Comment="" Name="ID" Type="WSTRING"/>
    </InputVars>
    <OutputVars>
      <VarDeclaration Comment="" Name="QO" Type="BOOL"/>
      <VarDeclaration Comment="" Name="STATUS" Type="WSTRING"/>
      <VarDeclaration Comment="" Name="RD_1" Type="ANY"/>
      <VarDeclaration Comment="" Name="RD_2" Type="ANY"/>
      <VarDeclaration Comment="" Name="RD_3" Type="ANY"/>
    </OutputVars>
  </InterfaceList>
</FBType>
```

Listing 7.65: IEC 61499 SUBSCRIBE_3 Function Block

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FBType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<ResourceType Comment="Most basic resource for executing FB networks" Name="EMB_RES">
  <Identification Description="Copyright (c) 2017 fortiss GmbH&#13;&#10; &#13;&#10;This program and
       the accompanying materials are made&#13;&#10;available under the terms of the Eclipse Public
       License 2.0&#13;&#10;which is available at https://www.eclipse.org/legal/epl
       -2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0"
    Function="This resource provides the most basic functionality of an IEC 61499 resource, namely
         executing FB networks. For convenience it already contains an instance of the E_RESTART FB
         providing events for starting up and shutting down an application.&#13;&#10;&#13;&#10;This
         resource is based on examples found in the different parts of IEC 61499 and the
         documentation at http://www.holobloc.com/doc/fb/rt/EMB_RES.htm"/>
  <VersionInfo Author="Alois Zoitl" Date="2017-12-02" Organization="fortiss GmbH" Version="1.0"/>
  <FBNetwork>
    <FB Comment="" Name="START" Type="E_RESTART" x="100.0" y="0.0"/>
  </FBNetwork>
</ResourceType>
```

Listing 7.66: IEC 61499 EMB_RES Resource

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DeviceType SYSTEM "http://www.holobloc.com/xml/Segment.dtd" >
<SegmentType Name="Ethernet" Comment="Ethernet Segment" >
  <Identification Standard="61499 ITA" Description="Copyright (c) 2008 Profactor GmbH&#13;&#10;
       &#13;&#10;This program and the accompanying materials are made&#13;&#10;available under the
       terms of the Eclipse Public License 2.0&#13;&#10;which is available at https://www.eclipse.
       org/legal/epl-2.0/&#13;&#10;&#13;&#10;SPDX-License-Identifier: EPL-2.0" />
  <VersionInfo Organization="Profactor GmbH" Version="0.1" Author="Gerhard Ebenhofer" Date="
       2008-06-02" Remarks="Initial version"/>
  <CompilerInfo/>
```

```
</SegmentType>
```

Listing 7.67: IEC 61499 Ethernet Segment

7.5 Framework Input Files

7.5.1 Substation Configuration Language File

```
<SCL xmlns="http://www.iec.ch/61850/2003/SCL"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    version="2007" revision="A"
>
  <Header id="SSD Example" />
  <Substation name="SS">
    <PowerTransformer name="T1" type="PTR">
      <TransformerWinding name="W1" type="PTW">
        <Terminal
           connectivityNode="SS/M/Q1/L"
           substationName="SS"
           voltageLevelName="M"
           bayName="Q1"
           cNodeName="L1"/>
      </TransformerWinding>
      <TransformerWinding name="W2" type="PTW">
        <Terminal
           connectivityNode="SS/L/Q4/L"
           substationName="SS"
           voltageLevelName="L"
           bayName="Q4"
           cNodeName="L1"/>
      </TransformerWinding>
    </PowerTransformer>
    <PowerTransformer name="T2" type="PTR">
      <TransformerWinding name="W1" type="PTW">
        <Terminal
           connectivityNode="SS/M/Q2/L"
           substationName="example"
           voltageLevelName="M"
           bayName="Q2"
           cNodeName="L1"/>
      </TransformerWinding>
      <TransformerWinding name="W2" type="PTW">
        <Terminal
           connectivityNode="SS/L/Q5/L"
           substationName="example"
           voltageLevelName="L"
           bayName="Q5"
           cNodeName="L1"/>
      </TransformerWinding>
    </PowerTransformer>
    <PowerTransformer name="T3" type="PTR">
      <TransformerWinding name="W1" type="PTW">
        <Terminal
           connectivityNode="SS/M/Q3/L"
           substationName="example"
           voltageLevelName="M"
           bayName="Q3"
           cNodeName="L1"/>
      </TransformerWinding>
      <TransformerWinding name="W2" type="PTW">
        <Terminal
           connectivityNode="SS/L/Q6/L"
           substationName="example"
           voltageLevelName="L"
           bayName="Q6"
           cNodeName="L1"/>
      </TransformerWinding>
    </PowerTransformer>
    <!-- Medium Voltage -->
    <VoltageLevel name="M">
```

```
  <Voltage multiplier="k" unit="V">50</Voltage>
  <Bay name="Q1">
    <ConnectivityNode name="L" pathName="SS/M/Q1/L"/>
  </Bay>
  <Bay name="Q2">
    <ConnectivityNode name="L" pathName="SS/M/Q2/L"/>
  </Bay>
  <Bay name="Q3">
    <ConnectivityNode name="L" pathName="SS/M/Q3/L"/>
  </Bay>
 </VoltageLevel>
 <!-- Low Voltage -->
 <VoltageLevel name="L">
  <Voltage unit="V">400</Voltage>
  <!-- static busbar -->
  <Bay name="Q4">
    <ConnectivityNode name="L" pathName="SS/L/Q4/L"/>
  </Bay>
  <!-- static busbar -->
  <Bay name="Q5">
    <ConnectivityNode name="L" pathName="SS/L/Q5/L"/>
  </Bay>
  <!-- static busbar -->
  <Bay name="Q6">
    <ConnectivityNode name="L" pathName="SS/L/Q6/L"/>
  </Bay>
  <Bay name="Q7">
    <ConductingEquipment name="QA" type="CBR">
      <LNode iedName="SSLQ7IED1" ldInst="LD1" lnClass="XCBR" lnInst="1" />
      <Terminal
        connectivityNode="SS/L/Q4/L"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q4"
        cNodeName="L1"/>
      <Terminal
        connectivityNode="SS/L/Q7/L1"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L1"/>
    </ConductingEquipment>
    <ConductingEquipment name="QB1" type="DIS">
      <LNode iedName="SSLQ7IED2" ldInst="LD1" lnClass="CSWI" lnInst="1" />
      <Terminal
        connectivityNode="SS/L/Q7/L1"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L1"/>
      <Terminal
        connectivityNode="SS/L/Q7/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L2"/>
    </ConductingEquipment>
    <ConductingEquipment name="QB2" type="DIS">
      <LNode iedName="SSLQ7IED3" ldInst="LD1" lnClass="CSWI" lnInst="1" />
      <Terminal
        connectivityNode="SS/L/Q7/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L2"/>
      <Terminal
        connectivityNode="SS/L/Q7/L3"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L3"/>
    </ConductingEquipment>
    <ConnectivityNode name="L1" pathName="SS/L/Q7/L1"/>
```

```
  <ConnectivityNode name="L2" pathName="SS/L/Q7/L2"/>
  <ConnectivityNode name="L3" pathName="SS/L/Q7/L3"/>
 </Bay>
 <Bay name="Q8">
  <ConductingEquipment name="QA" type="CBR">
   <Terminal
    connectivityNode="SS/L/Q5/L"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q5"
    cNodeName="L1"/>
   <Terminal
    connectivityNode="SS/L/Q8/L1"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q8"
    cNodeName="L1"/>
  </ConductingEquipment>
  <ConductingEquipment name="QB1" type="DIS">
   <Terminal
    connectivityNode="SS/L/Q8/L1"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q8"
    cNodeName="L1"/>
   <Terminal
    connectivityNode="SS/L/Q8/L2"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q8"
    cNodeName="L2"/>
  </ConductingEquipment>
  <ConductingEquipment name="QB2" type="DIS">
   <Terminal
    connectivityNode="SS/L/Q8/L2"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q8"
    cNodeName="L2"/>
   <Terminal
    connectivityNode="SS/L/Q8/L3"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q8"
    cNodeName="L3"/>
  </ConductingEquipment>
  <ConnectivityNode name="L1" pathName="SS/L/Q8/L1"/>
  <ConnectivityNode name="L2" pathName="SS/L/Q8/L2"/>
  <ConnectivityNode name="L3" pathName="SS/L/Q8/L3"/>
 </Bay>
 <Bay name="Q9">
  <ConductingEquipment name="QA" type="CBR">
   <Terminal
    connectivityNode="SS/L/Q6/L"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q6"
    cNodeName="L1"/>
   <Terminal
    connectivityNode="SS/L/Q9/L1"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q9"
    cNodeName="L1"/>
  </ConductingEquipment>
  <ConductingEquipment name="QB1" type="DIS">
   <Terminal
    connectivityNode="SS/L/Q9/L1"
    substationName="SS"
    voltageLevelName="L"
    bayName="Q9"
    cNodeName="L1"/>
   <Terminal
```

```
        connectivityNode="SS/L/Q9/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q9"
        cNodeName="L2"/>
      </ConductingEquipment>
      <ConductingEquipment name="QB2" type="DIS">
        <Terminal
        connectivityNode="SS/L/Q9/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q9"
        cNodeName="L2"/>
        <Terminal
        connectivityNode="SS/L/Q9/L3"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q9"
        cNodeName="L3"/>
      </ConductingEquipment>
      <ConnectivityNode name="L1" pathName="SS/L/Q9/L1"/>
      <ConnectivityNode name="L2" pathName="SS/L/Q9/L2"/>
      <ConnectivityNode name="L3" pathName="SS/L/Q9/L3"/>
    </Bay>
    <Bay name="Q10">
      <ConductingEquipment name="QB" type="DIS">
        <Terminal
        connectivityNode="SS/L/Q7/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q7"
        cNodeName="L2"/>
        <Terminal
        connectivityNode="SS/L/Q8/L2"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q8"
        cNodeName="L2"/>
      </ConductingEquipment>
    </Bay>
    <Bay name="Q11">
      <ConductingEquipment name="QB" type="DIS">
        <Terminal
        connectivityNode="SS/L/Q8/L3"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q8"
        cNodeName="L3"/>
        <Terminal
        connectivityNode="SS/L/Q9/L3"
        substationName="SS"
        voltageLevelName="L"
        bayName="Q9"
        cNodeName="L3"/>
      </ConductingEquipment>
    </Bay>
  </VoltageLevel>
</Substation>

<Communication>
  <SubNetwork type="IP" name="Mgmt">
    <Text>Management bus</Text>
    <ConnectedAP iedName="HMI" apName="S1" ></ConnectedAP>
    <ConnectedAP iedName="Router" apName="S2" ></ConnectedAP>
  </SubNetwork>
  <SubNetwork type="8-MMS" name="Station">
    <Text>Station bus</Text>
    <BitRate unit="b/s">10</BitRate>
    <!--<ConnectedAP iedName="Router" apName="S1" ></ConnectedAP>-->
    <ConnectedAP iedName="SSLQ7IED1" apName="S1" >
       <Address>
          <P type="IP">10.0.0.131</P>
          <P type="IP-SUBNET">255.255.255.0</P>
```

```
            <P type="MAC-Address">02-42-c0-a8-84-21</P>
         </Address>
         <GSE ldInst="LD1" cbName="gcbEventsMmxuMaxAPhs">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
      </ConnectedAP>
      <ConnectedAP iedName="SSLQ7IED2" apName="S1" >
         <Address>
           <P type="IP">10.0.0.132</P>
           <P type="IP-SUBNET">255.255.255.0</P>
           <P type="MAC-Address">02-42-c0-a8-84-22</P>
        </Address>
         <GSE ldInst="LD1" cbName="gcbEventsMmxuMaxAPhs">
         <Address>
            <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
            <P type="APPID">1000</P>
         </Address>
         <MinTime>1000</MinTime>
         <MaxTime>3000</MaxTime>
        </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
         <Address>
            <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
            <P type="APPID">1000</P>
         </Address>
         <MinTime>1000</MinTime>
         <MaxTime>3000</MaxTime>
        </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
         <Address>
            <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
            <P type="APPID">1000</P>
         </Address>
         <MinTime>1000</MinTime>
         <MaxTime>3000</MaxTime>
        </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
         <Address>
            <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
            <P type="APPID">1000</P>
         </Address>
         <MinTime>1000</MinTime>
         <MaxTime>3000</MaxTime>
        </GSE>
      </ConnectedAP>
      <ConnectedAP iedName="SSLQ7IED3" apName="S1" >
         <Address>
            <P type="IP">10.0.0.133</P>
            <P type="IP-SUBNET">255.255.255.0</P>
            <P type="MAC-Address">02-42-c0-a8-84-23</P>
         </Address>
         <GSE ldInst="LD1" cbName="gcbEventsPtrcTr">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
```

```
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
      </ConnectedAP>
      <ConnectedAP iedName="SSLQ7IED4" apName="S1" >
         <Address>
            <P type="IP">10.0.0.134</P>
            <P type="IP-SUBNET">255.255.255.0</P>
            <P type="MAC-Address">02-42-c0-a8-84-24</P>
         </Address>
         <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
      </ConnectedAP>
   </SubNetwork>
</Communication>

<IED name="SSLQ7IED1" type="SN0001" manufacturer="ABB">
   <Services>
      <DynAssociation />
      <GetDirectory />
      <GetDataObjectDefinition />
      <GetDataSetValue />
      <DataSetDirectory />
      <ReadWrite />
      <GetCBValues />
      <ConfLNs fixPrefix="true" fixLnInst="true" />
      <GOOSE max="5" />
      <GSSE max="5" />
      <FileHandling />
      <GSEDir />
      <TimerActivatedControl />
   </Services>
   <AccessPoint name="S1">
      <Server>
         <Authentication none="true" />
         <LDevice inst="LD1">
            <LN0 inst="" lnClass="LLN0" lnType="LN0">
               <DataSet name="EventsMmxuMaxAPhs" desc="EventsMmxuMaxAPhs">
                  <FCDA ldInst="LD1" lnClass="MMXU" fc="MX" lnInst="1" doName="MaxAPhs"/>
```

```
                    </DataSet>
                    <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                       <FCDA ldInst="LD1" lnClass="XCBR" fc="ST" lnInst="1" doName="Pos"/>
                    </DataSet>
                    <ReportControl name="EventsMmxuMaxAPhsRCB" confRev="1" datSet="EventsMmxuMaxAPhs"
                         rptID="EventsMmxuMaxAPhs" buffered="false" intgPd="1000" bufTime="50">
                       <TrgOps period="true" />
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                       <RptEnabled max="1" />
                    </ReportControl>
                    <ReportControl name="EventsXcbrPosRCB" confRev="1" datSet="EventsXcbrPos" rptID="
                         EventsXcbrPos" buffered="false" intgPd="1000" bufTime="50">
                       <TrgOps period="true" />
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                       <RptEnabled max="1" />
                    </ReportControl>
                    <GSEControl appID="eventsMmxuMaxAPhs" name="gcbEventsMmxuMaxAPhs" type="GOOSE"
                         datSet="EventsMmxuMaxAPhs" confRev="2" />
                    <GSEControl appID="eventsXcbrPos" name="gcbEventsXcbrPos" type="GOOSE" datSet="
                         EventsXcbrPos" confRev="2" />
                 </LN0>
                 <LN inst="1" lnClass="MMXU" lnType="MMXUa" />
                 <LN inst="1" lnClass="XCBR" lnType="XCBRa" />
              </LDevice>
           </Server>
        </AccessPoint>
     </IED>

     <IED name="SSLQ7IED2" type="SN0002" manufacturer="Siemens">
        <Services>
           <DynAssociation />
           <GetDirectory />
           <GetDataObjectDefinition />
           <GetDataSetValue />
           <DataSetDirectory />
           <ReadWrite />
           <GetCBValues />
           <ConfLNs fixPrefix="true" fixLnInst="true" />
           <GOOSE max="5" />
           <GSSE max="5" />
           <FileHandling />
           <GSEDir />
           <TimerActivatedControl />
        </Services>
        <AccessPoint name="S1">
           <Server>
              <Authentication none="true" />
              <LDevice inst="LD1">
                 <LN0 inst="" lnClass="LLN0" lnType="LN0">
                    <DataSet name="EventsMmxuMaxAPhs" desc="EventsMmxuMaxAPhs">
                       <FCDA ldInst="LD1" lnClass="MMXU" fc="MX" lnInst="1" doName="MaxAPhs"/>
                    </DataSet>
                    <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                       <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Str" />
                    </DataSet>
                    <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                       <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Op"/>
                    </DataSet>
                    <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                       <FCDA ldInst="LD1" lnClass="XCBR" fc="ST" lnInst="1" doName="Pos"/>
                    </DataSet>
                    <ReportControl name="EventsMmxuMaxAPhsRCB" confRev="1" datSet="EventsMmxuMaxAPhs"
                         rptID="EventsMmxuMaxAPhs" buffered="false" intgPd="1000" bufTime="50">
                       <TrgOps period="true" />
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                       <RptEnabled max="1" />
                    </ReportControl>
                    <ReportControl name="EventsPiocStrRCB" confRev="1" datSet="EventsPiocStr" rptID="
                         EventsPiocStr" buffered="false" intgPd="1000" bufTime="50">
                       <TrgOps period="true" />
```

```
                <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                    configRef="true" dataRef="true" entryId="true" />
                <RptEnabled max="1" />
             </ReportControl>
             <ReportControl name="EventsPiocOpRCB" confRev="1" datSet="EventsPiocOp" rptID="
                 EventsPiocOp" buffered="false" intgPd="1000" bufTime="50">
                <TrgOps period="true" />
                <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                    configRef="true" dataRef="true" entryId="true" />
                <RptEnabled max="1" />
             </ReportControl>
             <ReportControl name="EventsXcbrPosRCB" confRev="1" datSet="EventsXcbrPos" rptID="
                 EventsXcbrPos" buffered="false" intgPd="1000" bufTime="50">
                <TrgOps period="true" />
                <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                    configRef="true" dataRef="true" entryId="true" />
                <RptEnabled max="1" />
             </ReportControl>
             <GSEControl appID="eventsMmxuMaxAPhs" name="gcbEventsMmxuMaxAPhs" type="GOOSE"
                 datSet="EventsMmxuMaxAPhs" confRev="2" />
             <GSEControl appID="eventsPiocStr" name="gcbEventsPiocStr" type="GOOSE" datSet="
                 EventsPiocStr" confRev="2" />
             <GSEControl appID="eventsPiocOp" name="gcbEventsPiocOp" type="GOOSE" datSet="
                 EventsPiocOp" confRev="2" />
             <GSEControl appID="eventsXcbrPos" name="gcbEventsXcbrPos" type="GOOSE" datSet="
                 EventsXcbrPos" confRev="2" />
          </LN0>
          <LN inst="1" lnClass="MMXU" lnType="MMXUa" />
          <LN inst="1" lnClass="PIOC" lnType="PIOCa" />
          <LN inst="1" lnClass="XCBR" lnType="XCBRa" />
       </LDevice>
    </Server>
 </AccessPoint>
</IED>

<IED name="SSLQ7IED3" type="SN0003" manufacturer="Siemens">
   <Private type="Programmable">true</Private>
   <Private type="Memory">10000</Private>
   <Services>
      <DynAssociation />
      <GetDirectory />
      <GetDataObjectDefinition />
      <GetDataSetValue />
      <DataSetDirectory />
      <ReadWrite />
      <GetCBValues />
      <ConfLNs fixPrefix="true" fixLnInst="true" />
      <GOOSE max="5" />
      <GSSE max="5" />
      <FileHandling />
      <GSEDir />
      <TimerActivatedControl />
   </Services>
   <AccessPoint name="S1">
      <Server>
         <Authentication none="true" />
         <LDevice inst="LD1">
            <LN0 inst="" lnClass="LLN0" lnType="LN0">
               <DataSet name="EventsPtrcTr" desc="EventsPtrcTr">
                  <FCDA ldInst="LD1" lnClass="PTRC" fc="ST" lnInst="1" doName="Tr"/>
               </DataSet>
               <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                  <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Str" />
               </DataSet>
               <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                  <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Op"/>
               </DataSet>
               <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                  <FCDA ldInst="LD1" lnClass="XCBR" fc="ST" lnInst="1" doName="Pos"/>
               </DataSet>
               <ReportControl name="EventsPtrcTrRCB" confRev="1" datSet="EventsPtrcTr" rptID="
                   EventsPtrcTr" buffered="false" intgPd="1000" bufTime="50">
                  <TrgOps period="true" />
```

```
                        <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                        <RptEnabled max="1" />
                    </ReportControl>
                    <ReportControl name="EventsPiocStrRCB" confRev="1" datSet="EventsPiocStr" rptID="
                        EventsPiocStr" buffered="false" intgPd="1000" bufTime="50">
                        <TrgOps period="true" />
                        <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                        <RptEnabled max="1" />
                    </ReportControl>
                    <ReportControl name="EventsPiocOpRCB" confRev="1" datSet="EventsPiocOp" rptID="
                        EventsPiocOp" buffered="false" intgPd="1000" bufTime="50">
                        <TrgOps period="true" />
                        <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                        <RptEnabled max="1" />
                    </ReportControl>
                    <ReportControl name="EventsXcbrPosRCB" confRev="1" datSet="EventsXcbrPos" rptID="
                        EventsXcbrPos" buffered="false" intgPd="1000" bufTime="50">
                        <TrgOps period="true" />
                        <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            configRef="true" dataRef="true" entryId="true" />
                        <RptEnabled max="1" />
                    </ReportControl>
                    <GSEControl appID="eventsPtrcTr" name="gcbEventsPtrcTr" type="GOOSE" datSet="
                        EventsPtrcTr" confRev="2" />
                    <GSEControl appID="eventsPiocStr" name="gcbEventsPiocStr" type="GOOSE" datSet="
                        EventsPiocStr" confRev="2" />
                    <GSEControl appID="eventsPiocOp" name="gcbEventsPiocOp" type="GOOSE" datSet="
                        EventsPiocOp" confRev="2" />
                    <GSEControl appID="eventsXcbrPos" name="gcbEventsXcbrPos" type="GOOSE" datSet="
                        EventsXcbrPos" confRev="2" />
                </LN0>
                <LN inst="1" lnClass="PTRC" lnType="PTRCa" />
                <LN inst="1" lnClass="PIOC" lnType="PIOCa" />
                <LN inst="1" lnClass="XCBR" lnType="XCBRa" />
            </LDevice>
        </Server>
    </AccessPoint>
</IED>

<IED name="SSLQ7IED4" type="SN0004" manufacturer="Schneider">
    <Private type="Programmable">true</Private>
    <Private type="Memory">10000</Private>
    <Services>
        <DynAssociation />
        <GetDirectory />
        <GetDataObjectDefinition />
        <GetDataSetValue />
        <DataSetDirectory />
        <ReadWrite />
        <GetCBValues />
        <ConfLNs fixPrefix="true" fixLnInst="true" />
        <GOOSE max="5" />
        <GSSE max="5" />
        <FileHandling />
        <GSEDir />
        <TimerActivatedControl />
    </Services>
    <AccessPoint name="S1" router="true">
        <Server>
            <LDevice inst="LD1">
                <LN0 inst="" lnClass="LLN0" lnType="LN0">
                    <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                        <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Str" />
                    </DataSet>
                    <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                        <FCDA ldInst="LD1" lnClass="PIOC" fc="ST" lnInst="1" doName="Op"/>
                    </DataSet>
                    <ReportControl name="EventsPiocStrRCB" confRev="1" datSet="EventsPiocStr" rptID="
                        EventsPiocStr" buffered="false" intgPd="1000" bufTime="50">
                        <TrgOps period="true" />
```

```
                <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                    configRef="true" dataRef="true" entryId="true" />
                <RptEnabled max="1" />
             </ReportControl>
             <ReportControl name="EventsPiocOpRCB" confRev="1" datSet="EventsPiocOp" rptID="
                 EventsPiocOp" buffered="false" intgPd="1000" bufTime="50">
                <TrgOps period="true" />
                <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                    configRef="true" dataRef="true" entryId="true" />
                <RptEnabled max="1" />
             </ReportControl>
             <GSEControl appID="eventsPiocStr" name="gcbEventsPiocStr" type="GOOSE" datSet="
                 EventsPiocStr" confRev="2" />
             <GSEControl appID="eventsPiocOp" name="gcbEventsPiocOp" type="GOOSE" datSet="
                 EventsPiocOp" confRev="2" />
          </LN0>
          <LN inst="1" lnClass="PIOC" lnType="PIOCa" />
       </LDevice>
    </Server>
  </AccessPoint>
  <AccessPoint name="S2">
    <ServerAt apName="S2"/>
  </AccessPoint>
</IED>

<DataTypeTemplates>
  <LNodeType id="LN0" lnClass="LLN0">
    <DO name="Mod" type="myMod"/>
    <DO name="Beh" type="myBeh"/>
    <DO name="Health" type="myHealth"/>
    <DO name="NamPlt" type="myLN0LPL"/>
  </LNodeType>
  <LNodeType id="LPHDa" lnClass="LPHD">
    <DO name="PhyNam" type="myDPL"/>
    <DO name="PhyHealth" type="myINS"/>
    <DO name="Proxy" type="mySPS"/>
  </LNodeType>
  <LNodeType id="XCBRa" lnClass="XCBR">
    <DO name="Beh" type="myBeh"/>
    <DO name="LOC" type="mySPS"/>
    <DO name="OpCnt" type="myINS"/>
    <DO name="Pos" type="myPos"/>
    <DO name="BlkOpn" type="mySPC"/>
    <DO name="BlkCls" type="mySPC"/>
  </LNodeType>
  <LNodeType id="MMXUa" lnClass="MMXU">
    <DO name="Beh" type="myBeh" />
    <DO name="TotW" type="myMV"/>
    <DO name="TotVAr" type="myMV"/>
    <DO name="TotVA" type="myMV"/>
    <DO name="Hz" type="myMV"/>
    <DO name="MaxAPhs" type="myMV"/>
  </LNodeType>
  <LNodeType id="PIOCa" lnClass="PIOC">
    <DO name="Beh" type="myBeh" />
    <DO name="Str" type="myACD"/>
    <DO name="Op" type="myACT"/>
    <DO name="StrVal" type="myASG"/>
  </LNodeType>
  <LNodeType id="PTRCa" lnClass="PTRC">
    <DO name="Beh" type="myBeh" />
    <DO name="Str" type="myACD"/>
    <DO name="Op" type="myACT"/>
    <DO name="Tr" type="myACT"/>
  </LNodeType>

  <DOType id="myMod" cdc="INC">
    <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="Mod" />
    <DA name="q" fc="ST" qchg="true" bType="Quality" />
    <DA name="t" fc="ST" bType="Timestamp" />
    <DA name="ctlModel" fc="CF" bType="Enum" type="ctlModel" />
    <DA name="Oper" fc="CO" bType="Struct" type="myModOper" />
  </DOType>
```

```
<DOType id="myBeh" cdc="INS">
   <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="Beh" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myINS" cdc="INS">
   <DA name="stVal" fc="ST" dchg="true" bType="INT32" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myHealth" cdc="INS">
   <DA name="stVal" fc="ST" dchg="true" bType="Enum" type="Health" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myLN0LPL" cdc="LPL">
   <DA name="vendor" fc="DC" bType="VisString255"/>
   <DA name="swRev" fc="DC" bType="VisString255"/>
   <DA name="d" fc="DC" bType="VisString255" />
   <DA name="configRev" fc="DC" bType="VisString255"/>
   <DA name="ldNs" fc="EX" bType="VisString255"/>
</DOType>
<DOType id="myPos" cdc="DPC">
   <DA name="stVal" fc="ST" dchg="true" bType="Dbpos" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
   <DA name="ctlModel" fc="CF" bType="Enum" type="ctlModel" />
   <DA name="Oper" fc="CO" bType="Struct" type="myOper" />
   <DA name="SBOw" fc="CO" bType="Struct" type="myOper" />
   <DA name="Cancel" fc="CO" bType="Struct" type="myCancel" />
</DOType>
<DOType id="mySPS" cdc="SPS">
   <DA name="stVal" fc="ST" dchg="true" bType="BOOLEAN" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myMV" cdc="MV">
   <DA name="mag" fc="MX" dchg="true" bType="Struct" type="myAnalogValue" />
   <DA name="q" fc="MX" qchg="true" bType="Quality" />
   <DA name="t" fc="MX" bType="Timestamp" />
   <DA name="sVC" fc="CF" dchg="true" bType="Struct" type="ScaledValueConfig" />
</DOType>
<DOType id="myACD" cdc="ACD">
   <DA name="general" fc="ST" dchg="true" bType="BOOLEAN" />
   <DA name="dirGeneral" fc="ST" dchg="true" bType="Enum" type="ACDdir" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myACT" cdc="ACT">
   <DA name="general" fc="ST" dchg="true" bType="BOOLEAN" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myASG" cdc="ASG">
   <DA name="setMag" fc="SP" dchg="true" bType="Struct" type="myAnalogValue" />
   <DA name="units" fc="CF" dchg="true" bType="Enum" type="SIUnit" />
   <DA name="sVC" fc="CF" dchg="true" bType="Struct" type="ScaledValueConfig" />
</DOType>
<DOType id="mySPC" cdc="SPC">
   <DA name="stVal" fc="ST" dchg="true" bType="BOOLEAN" />
   <DA name="q" fc="ST" qchg="true" bType="Quality" />
   <DA name="t" fc="ST" bType="Timestamp" />
</DOType>
<DOType id="myDPL" cdc="DPL">
   <DA name="vendor" fc="DC" bType="VisString255"/>
   <DA name="hwRev" fc="DC" bType="VisString255"/>
</DOType>

<DAType id="myAnalogValue">
   <BDA name="f" bType="FLOAT32" />
</DAType>
<DAType id="ScaledValueConfig">
   <BDA name="scaleFactor" bType="FLOAT32" />
```

```
    <BDA name="offset" bType="FLOAT32" />
  </DAType>
  <DAType id="myModOper">
    <BDA name="ctlVal" bType="Enum" type="Mod" />
    <BDA name="origin" bType="Struct" type="originator" />
    <BDA name="ctlNum" bType="INT8U" />
    <BDA name="T" bType="Timestamp" />
    <BDA name="Test" bType="BOOLEAN" />
    <BDA name="Check" bType="Check" />
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
  </DAType>
  <DAType id="myOper">
    <BDA name="ctlVal" bType="BOOLEAN" />
    <BDA name="origin" bType="Struct" type="originator" />
    <BDA name="ctlNum" bType="INT8U" />
    <BDA name="T" bType="Timestamp" />
    <BDA name="Test" bType="BOOLEAN" />
    <BDA name="Check" bType="Check" />
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
  </DAType>
  <DAType id="myCancel">
    <BDA name="ctlVal" bType="BOOLEAN" />
    <BDA name="origin" bType="Struct" type="originator" />
    <BDA name="ctlNum" bType="INT8U" />
    <BDA name="T" bType="Timestamp" />
    <BDA name="Test" bType="BOOLEAN" />
    <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
  </DAType>
  <DAType id="originator">
    <BDA name="orCat" cond="M" bType="Enum" type="orCategory"/>
    <BDA name="orIdent" cond="M" bType="Octet64"/>
  </DAType>

  <EnumType id="ACDdir">
    <EnumVal ord="0">unknown</EnumVal>
    <EnumVal ord="1">forward</EnumVal>
    <EnumVal ord="2">backward</EnumVal>
    <EnumVal ord="3">both</EnumVal>
  </EnumType>
  <EnumType id="Health">
    <EnumVal ord="1">Ok</EnumVal>
    <EnumVal ord="2">Warning</EnumVal>
    <EnumVal ord="3">Alarm</EnumVal>
  </EnumType>
  <EnumType id="ctlModel">
    <EnumVal ord="0">status-only</EnumVal>
    <EnumVal ord="1">direct-with-normal-security</EnumVal>
    <EnumVal ord="2">sbo-with-normal-security</EnumVal>
    <EnumVal ord="3">direct-with-enhanced-security</EnumVal>
    <EnumVal ord="4">sbo-with-enhanced-security</EnumVal>
  </EnumType>
  <EnumType id="orCategory">
    <EnumVal ord="0">not-supported</EnumVal>
    <EnumVal ord="1">bay-control</EnumVal>
    <EnumVal ord="2">station-control</EnumVal>
    <EnumVal ord="3">remote-control</EnumVal>
    <EnumVal ord="4">automatic-bay</EnumVal>
    <EnumVal ord="5">automatic-station</EnumVal>
    <EnumVal ord="6">automatic-remote</EnumVal>
    <EnumVal ord="7">maintenance</EnumVal>
    <EnumVal ord="8">process</EnumVal>
  </EnumType>
  <EnumType id="Beh">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
   <EnumVal ord="4">test/blocked</EnumVal>
   <EnumVal ord="5">off</EnumVal>
  </EnumType>
  <EnumType id="Mod">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
```

```
    <EnumVal ord="4">test/blocked</EnumVal>
    <EnumVal ord="5">off</EnumVal>
</EnumType>
    <EnumType id="Health">
    <EnumVal ord="1">Ok</EnumVal>
    <EnumVal ord="2">Warning</EnumVal>
    <EnumVal ord="3">Alarm</EnumVal>
</EnumType>
<EnumType id="SIUnit">
  <EnumVal ord="1"/>
  <EnumVal ord="2">m</EnumVal>
  <EnumVal ord="3">kg</EnumVal>
  <EnumVal ord="4">s</EnumVal>
  <EnumVal ord="5">A</EnumVal>
  <EnumVal ord="6">K</EnumVal>
  <EnumVal ord="7">mol</EnumVal>
  <EnumVal ord="8">cd</EnumVal>
  <EnumVal ord="9">deg</EnumVal>
  <EnumVal ord="10">rad</EnumVal>
  <EnumVal ord="11">sr</EnumVal>
  <EnumVal ord="21">Gy</EnumVal>
  <EnumVal ord="22">q</EnumVal>
  <EnumVal ord="23">°C</EnumVal>
  <EnumVal ord="24">Sv</EnumVal>
  <EnumVal ord="25">F</EnumVal>
  <EnumVal ord="26">C</EnumVal>
  <EnumVal ord="27">S</EnumVal>
  <EnumVal ord="28">H</EnumVal>
  <EnumVal ord="29">V</EnumVal>
  <EnumVal ord="30">ohm</EnumVal>
  <EnumVal ord="31">J</EnumVal>
  <EnumVal ord="32">N</EnumVal>
  <EnumVal ord="33">Hz</EnumVal>
  <EnumVal ord="34">lx</EnumVal>
  <EnumVal ord="35">Lm</EnumVal>
  <EnumVal ord="36">Wb</EnumVal>
  <EnumVal ord="37">T</EnumVal>
  <EnumVal ord="38">W</EnumVal>
  <EnumVal ord="39">Pa</EnumVal>
  <EnumVal ord="41">m2</EnumVal>
  <EnumVal ord="42">m3</EnumVal>
  <EnumVal ord="43">m/s</EnumVal>
  <EnumVal ord="44">m/s2</EnumVal>
  <EnumVal ord="45">m3/s</EnumVal>
  <EnumVal ord="46">m/m3</EnumVal>
  <EnumVal ord="47">M</EnumVal>
  <EnumVal ord="48">kg/m3</EnumVal>
  <EnumVal ord="49">m2/s</EnumVal>
  <EnumVal ord="50">W/m K</EnumVal>
  <EnumVal ord="51">J/K</EnumVal>
  <EnumVal ord="52">ppm</EnumVal>
  <EnumVal ord="53">1/s</EnumVal>
  <EnumVal ord="54">rad/s</EnumVal>
  <EnumVal ord="61">VA</EnumVal>
  <EnumVal ord="62">Watts</EnumVal>
  <EnumVal ord="63">VAr</EnumVal>
  <EnumVal ord="64">phi</EnumVal>
  <EnumVal ord="65">cos(phi)</EnumVal>
  <EnumVal ord="66">Vs</EnumVal>
  <EnumVal ord="67">V2</EnumVal>
  <EnumVal ord="68">As</EnumVal>
  <EnumVal ord="69">A2</EnumVal>
  <EnumVal ord="70">A2t</EnumVal>
  <EnumVal ord="71">VAh</EnumVal>
  <EnumVal ord="72">Wh</EnumVal>
  <EnumVal ord="73">VArh</EnumVal>
  <EnumVal ord="74">V/Hz</EnumVal>
  <EnumVal ord="75">Hz/s</EnumVal>
  <EnumVal ord="76">char</EnumVal>
  <EnumVal ord="77">char/s</EnumVal>
  <EnumVal ord="78">kgm2</EnumVal>
  <EnumVal ord="79">dB</EnumVal>
</EnumType>
```

```
  </DataTypeTemplates>
</SCL>
```

Listing 7.68: IEC 61850 Substation Configuration Language File

7.5.2 System File

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE System SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd">
<System Comment="" Name="diplomarbeit">
  <VersionInfo Author="fxk14i" Date="2021-02-09" Version="1.0"/>
  <Application Comment="" Name="diplomarbeitApp">
    <SubAppNetwork>
      <FB Comment="" Name="iec_61850_MMXU" Type="iec_61850_MMXU" x="705.8823529411765" y="
          1647.0588235294117"/>
      <FB Comment="" Name="iec_61850_PTRC" Type="iec_61850_PTRC" x="3764.705882352941" y="
          117.6470588235294"/>
      <FB Comment="" Name="iec_61850_XCBR" Type="iec_61850_XCBR" x="4941.176470588235" y="
          117.6470588235294"/>
      <FB Comment="" Name="READ_FROM_DB" Type="READ_FROM_DB" x="3411.7647058823527" y="
          2470.5882352941176">
        <Parameter Name="ID" Value="10.0.0.130:5000"/>
      </FB>
      <FB Comment="" Name="STORE_IN_DB" Type="STORE_IN_DB" x="6117.647058823529" y="
          1647.0588235294117">
        <Parameter Name="ID" Value="10.0.0.130:5000"/>
      </FB>
      <FB Comment="" Name="SEND_SMS" Type="SEND_SMS" x="4470.588235294117" y="2470.5882352941176">
        <Parameter Name="ID" Value="10.0.0.130:5000"/>
      </FB>
      <FB Comment="" Name="E_CYCLE" Type="E_CYCLE" x="2470.5882352941176" y="2470.5882352941176">
        <Parameter Name="DT" Value="T#5s"/>
      </FB>
      <FB Comment="" Name="COMPOSE_ASG" Type="COMPOSE_ASG" x="823.5294117647059" y="470.5882352941176
          ">
        <Parameter Name="setMag" Value="( f := 12.0)"/>
        <Parameter Name="units" Value="( SIUnit := 5, multiplier := 0 )"/>
      </FB>
      <FB Comment="" Name="E_DELAY" Type="E_DELAY">
        <Parameter Name="DT" Value="T#10s"/>
      </FB>
      <FB Comment="" Name="DECOMPOSE_DPC" Type="DECOMPOSE_DPC" x="4941.176470588235" y="
          1647.0588235294117"/>
      <FB Comment="" Name="iec_61850_PIOC" Type="iec_61850_PIOC" x="2588.235294117647" y="
          117.6470588235294"/>
      <EventConnections>
        <Connection Comment="" Destination="COMPOSE_ASG.REQ" Source="E_DELAY.EO"/>
        <Connection Comment="" Destination="iec_61850_XCBR.REQ" Source="iec_61850_PTRC.TRG" dx1="
            311.7647058823529"/>
        <Connection Comment="" Destination="SEND_SMS.REQ" Source="READ_FROM_DB.CNF" dx1="
            282.35294117647055"/>
        <Connection Comment="" Destination="READ_FROM_DB.REQ" Source="E_CYCLE.EO" dx1="
            223.52941176470586"/>
        <Connection Comment="" Destination="STORE_IN_DB.REQ" Source="DECOMPOSE_DPC.CNF" dx1="
            211.76470588235293"/>
        <Connection Comment="" Destination="DECOMPOSE_DPC.REQ" Source="iec_61850_XCBR.TRG" dx1="
            223.52941176470586" dx2="70.58823529411764" dy="1035.2941176470588"/>
        <Connection Comment="" Destination="iec_61850_PIOC.REQ" Source="COMPOSE_ASG.CNF" dx1="
            417.6470588235294"/>
        <Connection Comment="" Destination="iec_61850_PIOC.REQ" Source="iec_61850_MMXU.TRG" dx1="
            411.7647058823529"/>
      </EventConnections>
      <DataConnections>
        <Connection Comment="" Destination="iec_61850_PIOC.asgVal" Source="COMPOSE_ASG.asgVal" dx1="
            541.1764705882352"/>
        <Connection Comment="" Destination="iec_61850_PIOC.mvVal" Source="iec_61850_MMXU.MaxAPhs" dx1
            ="641.1764705882352"/>
        <Connection Comment="" Destination="iec_61850_XCBR.Tr" Source="iec_61850_PTRC.Tr"/>
        <Connection Comment="" Destination="SEND_SMS.Str" Source="READ_FROM_DB.Str" dx1="
            282.35294117647055"/>
        <Connection Comment="" Destination="STORE_IN_DB.stVal" Source="DECOMPOSE_DPC.stVal" dx1="
            211.76470588235293"/>
```

```
      <Connection Comment="" Destination="STORE_IN_DB.q" Source="DECOMPOSE_DPC.q" dx1="
          211.76470588235293"/>
      <Connection Comment="" Destination="STORE_IN_DB.t" Source="DECOMPOSE_DPC.t" dx1="
          211.76470588235293"/>
      <Connection Comment="" Destination="DECOMPOSE_DPC.dpcVal" Source="iec_61850_XCBR.Pos_Out" dx1
          ="70.58823529411764" dx2="176.47058823529412" dy="447.0588235294117"/>
      <Connection Comment="" Destination="iec_61850_PTRC.Op_In" Source="iec_61850_PIOC.Op"/>
    </DataConnections>
  </SubAppNetwork>
 </Application>
 <Device Comment="" Name="TEMPLATE" Type="TEMPLATE" x="1400.0" y="1170.5882352941176">
  <Attribute Comment="device profile" Name="Profile" Type="STRING" Value="HOLOBLOC"/>
  <Attribute Comment="color" Name="Color" Type="STRING" Value="57,194,102"/>
  <Resource Comment="" Name="EMB_RES" Type="EMB_RES" x="0.0" y="0.0">
   <FBNetwork/>
  </Resource>
 </Device>
</System>
```

Listing 7.69: IEC 61499 System File

7.6 Framework Output Files

7.6.1 Mappings File

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Mappings SCL="res/at/ac/tuwien/diplomarbeit.scl" System="res/at/ac/tuwien/diplomarbeit.sys">
   <Mapping factor="0.327" id="3014e2ae-17b9-4937-b9c2-9d523abbde25">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="E_CYCLE"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.2552" id="0f3d1f74-50fd-4700-95e6-22392522406c">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.2458" id="8caa91e6-aa15-4c95-9039-8f1dc34f5c0f">
```

```
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="E_DELAY"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3885" id="9e9f302b-b292-4d0f-9ff7-8a8f6868c1a8">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.2462" id="7530a393-d703-42b8-936b-5cbd3a06ceb6">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="COMPOSE_ASG"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3889" id="ede57c10-52ba-4a1d-a41f-0a08d50f6de6">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="E_DELAY"/>
      </IED>
```

```
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3795" id="9f5bcef7-0442-43a9-81eb-00df892e2b6d">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3951" id="7e53e13c-e950-4c29-9a0c-45d58acfa392">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="SEND_SMS"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="E_CYCLE"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3433" id="d4cc2ac5-f6be-49d6-9c33-8d2c5aeb2529">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="E_CYCLE"/>
         <FB name="COMPOSE_ASG"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="SEND_SMS"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
```

```
   <Mapping factor="0.3437" id="14937307-6b0b-4ad9-8d70-39060bf837f6">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="E_CYCLE"/>
         <FB name="E_DELAY"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.3343" id="396e7d75-9adf-49f1-8c04-840d626954f7">
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="E_CYCLE"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
      </IED>
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
   </Mapping>
   <Mapping factor="0.4403" id="30fd8f08-4505-4ed0-b2a0-8c74a922d6f8">
      <IED name="SSLQ7IED1">
         <FB name="iec_61850_MMXU"/>
         <FB name="iec_61850_XCBR"/>
      </IED>
      <IED name="SSLQ7IED2">
         <FB name="iec_61850_PIOC"/>
      </IED>
      <IED name="SSLQ7IED3">
         <FB name="iec_61850_PTRC"/>
         <FB name="READ_FROM_DB"/>
         <FB name="STORE_IN_DB"/>
         <FB name="E_CYCLE"/>
      </IED>
      <IED name="SSLQ7IED4">
         <FB name="SEND_SMS"/>
         <FB name="COMPOSE_ASG"/>
         <FB name="E_DELAY"/>
         <FB name="DECOMPOSE_DPC"/>
      </IED>
   </Mapping>
   <!-- ... -->
</Mappings>
```

Listing 7.70: Framework Mappings File

7.6.2 Configured IED Description Files

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SCL xmlns="http://www.iec.ch/61850/2003/SCL" version="2007" revision="A">
```

```
<Header id="SSD Example"/>
<Communication>
   <SubNetwork type="IP" name="Mgmt" desc="">
      <Text>Management bus</Text>
   </SubNetwork>
   <SubNetwork type="8-MMS" name="Station" desc="">
      <Text>Station bus</Text>
      <BitRate unit="b/s">10</BitRate>
      <ConnectedAP iedName="SSLQ7IED1" apName="S1">
         <Address>
            <P type="IP">10.0.0.131</P>
            <P type="IP-SUBNET">255.255.255.0</P>
            <P type="MAC-Address">02-42-c0-a8-84-21</P>
         </Address>
         <GSE ldInst="LD1" cbName="gcbEventsMmxuMaxAPhs">
            <Address>
               <P type="MAC-Address">02-42-c0-a8-84-22</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
            <Address>
               <P type="MAC-Address">02-42-c0-a8-84-24</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
      </ConnectedAP>
   </SubNetwork>
</Communication>
<IED name="SSLQ7IED1" type="SN0001" manufacturer="ABB">
   <Services>
      <DynAssociation/>
      <GetDirectory/>
      <GetDataObjectDefinition/>
      <GetDataSetValue/>
      <DataSetDirectory/>
      <ReadWrite/>
      <TimerActivatedControl/>
      <GetCBValues/>
      <GSEDir/>
      <GOOSE max="5"/>
      <GSSE max="5"/>
      <FileHandling/>
      <ConfLNs fixPrefix="true" fixLnInst="true"/>
   </Services>
   <AccessPoint name="S1">
      <Server>
         <Authentication none="true"/>
         <LDevice inst="LD1">
            <LN0 lnClass="LLN0" inst="" lnType="LN0">
               <DataSet name="EventsMmxuMaxAPhs" desc="EventsMmxuMaxAPhs">
                  <FCDA ldInst="LD1" lnClass="MMXU" lnInst="1" doName="MaxAPhs" fc="MX"/>
               </DataSet>
               <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                  <FCDA ldInst="LD1" lnClass="XCBR" lnInst="1" doName="Pos" fc="ST"/>
               </DataSet>
               <ReportControl rptID="EventsMmxuMaxAPhs" confRev="1" buffered="false" bufTime="50
                    " intgPd="1000" name="EventsMmxuMaxAPhsRCB" datSet="EventsMmxuMaxAPhs">
                  <TrgOps period="true"/>
                  <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                       dataRef="true" configRef="true"/>
                  <RptEnabled max="1"/>
               </ReportControl>
               <ReportControl rptID="EventsXcbrPos" confRev="1" buffered="false" bufTime="50"
                    intgPd="1000" name="EventsXcbrPosRCB" datSet="EventsXcbrPos">
                  <TrgOps period="true"/>
                  <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                       dataRef="true" configRef="true"/>
                  <RptEnabled max="1"/>
```

```
            </ReportControl>
            <GSEControl type="GOOSE" appID="eventsMmxuMaxAPhs" confRev="2" name="
                gcbEventsMmxuMaxAPhs" datSet="EventsMmxuMaxAPhs"/>
            <GSEControl type="GOOSE" appID="eventsXcbrPos" confRev="2" name="gcbEventsXcbrPos
                " datSet="EventsXcbrPos"/>
         </LN0>
         <LN lnClass="MMXU" inst="1" lnType="MMXUa"/>
         <LN lnClass="XCBR" inst="1" lnType="XCBRa">
            <Inputs>
               <ExtRef iedName="SSLQ7IED3" ldInst="LD1" lnClass="PTRC" lnInst="1" doName="Tr"
                    srcLNClass="LLN0" srcCBName="gcbEventsPtrcTr"/>
            </Inputs>
         </LN>
      </LDevice>
   </Server>
</AccessPoint>
</IED>
<DataTypeTemplates>
   <LNodeType lnClass="LLN0" id="LN0">
      <DO name="Mod" type="myMod"/>
      <DO name="Beh" type="myBeh"/>
      <DO name="Health" type="myHealth"/>
      <DO name="NamPlt" type="myLN0LPL"/>
   </LNodeType>
   <LNodeType lnClass="LPHD" id="LPHDa">
      <DO name="PhyNam" type="myDPL"/>
      <DO name="PhyHealth" type="myINS"/>
      <DO name="Proxy" type="mySPS"/>
   </LNodeType>
   <LNodeType lnClass="XCBR" id="XCBRa">
      <DO name="Beh" type="myBeh"/>
      <DO name="LOC" type="mySPS"/>
      <DO name="OpCnt" type="myINS"/>
      <DO name="Pos" type="myPos"/>
      <DO name="BlkOpn" type="mySPC"/>
      <DO name="BlkCls" type="mySPC"/>
   </LNodeType>
   <LNodeType lnClass="MMXU" id="MMXUa">
      <DO name="Beh" type="myBeh"/>
      <DO name="TotW" type="myMV"/>
      <DO name="TotVAr" type="myMV"/>
      <DO name="TotVA" type="myMV"/>
      <DO name="Hz" type="myMV"/>
      <DO name="MaxAPhs" type="myMV"/>
   </LNodeType>
   <LNodeType lnClass="PIOC" id="PIOCa">
      <DO name="Beh" type="myBeh"/>
      <DO name="Str" type="myACD"/>
      <DO name="Op" type="myACT"/>
      <DO name="StrVal" type="myASG"/>
   </LNodeType>
   <LNodeType lnClass="PTRC" id="PTRCa">
      <DO name="Beh" type="myBeh"/>
      <DO name="Str" type="myACD"/>
      <DO name="Op" type="myACT"/>
      <DO name="Tr" type="myACT"/>
   </LNodeType>
   <DOType cdc="INC" id="myMod">
      <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Mod"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
      <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
      <DA fc="CO" name="Oper" bType="Struct" type="myModOper"/>
   </DOType>
   <DOType cdc="INS" id="myBeh">
      <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Beh"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
   </DOType>
   <DOType cdc="INS" id="myINS">
      <DA fc="ST" dchg="true" name="stVal" bType="INT32"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
```

```
</DOType>
<DOType cdc="INS" id="myHealth">
   <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Health"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="LPL" id="myLN0LPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="swRev" bType="VisString255"/>
   <DA fc="DC" name="d" bType="VisString255"/>
   <DA fc="DC" name="configRev" bType="VisString255"/>
   <DA fc="EX" name="ldNs" bType="VisString255"/>
</DOType>
<DOType cdc="DPC" id="myPos">
   <DA fc="ST" dchg="true" name="stVal" bType="Dbpos"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
   <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
   <DA fc="CO" name="Oper" bType="Struct" type="myOper"/>
   <DA fc="CO" name="SBOw" bType="Struct" type="myOper"/>
   <DA fc="CO" name="Cancel" bType="Struct" type="myCancel"/>
</DOType>
<DOType cdc="SPS" id="mySPS">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="MV" id="myMV">
   <DA fc="MX" dchg="true" name="mag" bType="Struct" type="myAnalogValue"/>
   <DA fc="MX" qchg="true" name="q" bType="Quality"/>
   <DA fc="MX" name="t" bType="Timestamp"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="ACD" id="myACD">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" dchg="true" name="dirGeneral" bType="Enum" type="ACDdir"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ACT" id="myACT">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ASG" id="myASG">
   <DA fc="SP" dchg="true" name="setMag" bType="Struct" type="myAnalogValue"/>
   <DA fc="CF" dchg="true" name="units" bType="Enum" type="SIUnit"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="SPC" id="mySPC">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="DPL" id="myDPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="hwRev" bType="VisString255"/>
</DOType>
<DAType id="myAnalogValue">
   <BDA name="f" bType="FLOAT32"/>
</DAType>
<DAType id="ScaledValueConfig">
   <BDA name="scaleFactor" bType="FLOAT32"/>
   <BDA name="offset" bType="FLOAT32"/>
</DAType>
<DAType id="myModOper">
   <BDA name="ctlVal" bType="Enum" type="Mod"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <BDA name="Check" bType="Check"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
```

```
</DAType>
<DAType id="myOper">
   <BDA name="ctlVal" bType="BOOLEAN"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <BDA name="Check" bType="Check"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="myCancel">
   <BDA name="ctlVal" bType="BOOLEAN"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="originator">
   <BDA name="orCat" bType="Enum" type="orCategory" cond="M"/>
   <BDA name="orIdent" bType="Octet64" cond="M"/>
</DAType>
<EnumType id="ACDdir">
   <EnumVal ord="0">unknown</EnumVal>
   <EnumVal ord="1">forward</EnumVal>
   <EnumVal ord="2">backward</EnumVal>
   <EnumVal ord="3">both</EnumVal>
</EnumType>
<EnumType id="Health">
   <EnumVal ord="1">Ok</EnumVal>
   <EnumVal ord="2">Warning</EnumVal>
   <EnumVal ord="3">Alarm</EnumVal>
</EnumType>
<EnumType id="ctlModel">
   <EnumVal ord="0">status-only</EnumVal>
   <EnumVal ord="1">direct-with-normal-security</EnumVal>
   <EnumVal ord="2">sbo-with-normal-security</EnumVal>
   <EnumVal ord="3">direct-with-enhanced-security</EnumVal>
   <EnumVal ord="4">sbo-with-enhanced-security</EnumVal>
</EnumType>
<EnumType id="orCategory">
   <EnumVal ord="0">not-supported</EnumVal>
   <EnumVal ord="1">bay-control</EnumVal>
   <EnumVal ord="2">station-control</EnumVal>
   <EnumVal ord="3">remote-control</EnumVal>
   <EnumVal ord="4">automatic-bay</EnumVal>
   <EnumVal ord="5">automatic-station</EnumVal>
   <EnumVal ord="6">automatic-remote</EnumVal>
   <EnumVal ord="7">maintenance</EnumVal>
   <EnumVal ord="8">process</EnumVal>
</EnumType>
<EnumType id="Beh">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
   <EnumVal ord="4">test/blocked</EnumVal>
   <EnumVal ord="5">off</EnumVal>
</EnumType>
<EnumType id="Mod">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
   <EnumVal ord="4">test/blocked</EnumVal>
   <EnumVal ord="5">off</EnumVal>
</EnumType>
<EnumType id="Health">
   <EnumVal ord="1">Ok</EnumVal>
   <EnumVal ord="2">Warning</EnumVal>
   <EnumVal ord="3">Alarm</EnumVal>
</EnumType>
<EnumType id="SIUnit">
   <EnumVal ord="1"></EnumVal>
   <EnumVal ord="2">m</EnumVal>
```

```
            <EnumVal ord="3">kg</EnumVal>
            <EnumVal ord="4">s</EnumVal>
            <EnumVal ord="5">A</EnumVal>
            <EnumVal ord="6">K</EnumVal>
            <EnumVal ord="7">mol</EnumVal>
            <EnumVal ord="8">cd</EnumVal>
            <EnumVal ord="9">deg</EnumVal>
            <EnumVal ord="10">rad</EnumVal>
            <EnumVal ord="11">sr</EnumVal>
            <EnumVal ord="21">Gy</EnumVal>
            <EnumVal ord="22">q</EnumVal>
            <EnumVal ord="23">°C</EnumVal>
            <EnumVal ord="24">Sv</EnumVal>
            <EnumVal ord="25">F</EnumVal>
            <EnumVal ord="26">C</EnumVal>
            <EnumVal ord="27">S</EnumVal>
            <EnumVal ord="28">H</EnumVal>
            <EnumVal ord="29">V</EnumVal>
            <EnumVal ord="30">ohm</EnumVal>
            <EnumVal ord="31">J</EnumVal>
            <EnumVal ord="32">N</EnumVal>
            <EnumVal ord="33">Hz</EnumVal>
            <EnumVal ord="34">lx</EnumVal>
            <EnumVal ord="35">Lm</EnumVal>
            <EnumVal ord="36">Wb</EnumVal>
            <EnumVal ord="37">T</EnumVal>
            <EnumVal ord="38">W</EnumVal>
            <EnumVal ord="39">Pa</EnumVal>
            <EnumVal ord="41">m2</EnumVal>
            <EnumVal ord="42">m3</EnumVal>
            <EnumVal ord="43">m/s</EnumVal>
            <EnumVal ord="44">m/s2</EnumVal>
            <EnumVal ord="45">m3/s</EnumVal>
            <EnumVal ord="46">m/m3</EnumVal>
            <EnumVal ord="47">M</EnumVal>
            <EnumVal ord="48">kg/m3</EnumVal>
            <EnumVal ord="49">m2/s</EnumVal>
            <EnumVal ord="50">W/m K</EnumVal>
            <EnumVal ord="51">J/K</EnumVal>
            <EnumVal ord="52">ppm</EnumVal>
            <EnumVal ord="53">1/s</EnumVal>
            <EnumVal ord="54">rad/s</EnumVal>
            <EnumVal ord="61">VA</EnumVal>
            <EnumVal ord="62">Watts</EnumVal>
            <EnumVal ord="63">VAr</EnumVal>
            <EnumVal ord="64">phi</EnumVal>
            <EnumVal ord="65">cos(phi)</EnumVal>
            <EnumVal ord="66">Vs</EnumVal>
            <EnumVal ord="67">V2</EnumVal>
            <EnumVal ord="68">As</EnumVal>
            <EnumVal ord="69">A2</EnumVal>
            <EnumVal ord="70">A2t</EnumVal>
            <EnumVal ord="71">VAh</EnumVal>
            <EnumVal ord="72">Wh</EnumVal>
            <EnumVal ord="73">VArh</EnumVal>
            <EnumVal ord="74">V/Hz</EnumVal>
            <EnumVal ord="75">Hz/s</EnumVal>
            <EnumVal ord="76">char</EnumVal>
            <EnumVal ord="77">char/s</EnumVal>
            <EnumVal ord="78">kgm2</EnumVal>
            <EnumVal ord="79">dB</EnumVal>
         </EnumType>
      </DataTypeTemplates>
   </SCL>
```

Listing 7.71: Configured IED Description File for IED 1

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SCL xmlns="http://www.iec.ch/61850/2003/SCL" version="2007" revision="A">
   <Header id="SSD Example"/>
   <Communication>
      <SubNetwork type="IP" name="Mgmt" desc="">
```

```
      <Text>Management bus</Text>
   </SubNetwork>
   <SubNetwork type="8-MMS" name="Station" desc="">
      <Text>Station bus</Text>
      <BitRate unit="b/s">10</BitRate>
      <ConnectedAP iedName="SSLQ7IED2" apName="S1">
         <Address>
            <P type="IP">10.0.0.132</P>
            <P type="IP-SUBNET">255.255.255.0</P>
            <P type="MAC-Address">02-42-c0-a8-84-22</P>
         </Address>
         <GSE ldInst="LD1" cbName="gcbEventsMmxuMaxAPhs">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
            <Address>
               <P type="MAC-Address">02-42-c0-a8-84-23</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
         <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
            <Address>
               <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
               <P type="APPID">1000</P>
            </Address>
            <MinTime>1000</MinTime>
            <MaxTime>3000</MaxTime>
         </GSE>
      </ConnectedAP>
   </SubNetwork>
</Communication>
<IED name="SSLQ7IED2" type="SN0002" manufacturer="Siemens">
   <Services>
      <DynAssociation/>
      <GetDirectory/>
      <GetDataObjectDefinition/>
      <GetDataSetValue/>
      <DataSetDirectory/>
      <ReadWrite/>
      <TimerActivatedControl/>
      <GetCBValues/>
      <GSEDir/>
      <GOOSE max="5"/>
      <GSSE max="5"/>
      <FileHandling/>
      <ConfLNs fixPrefix="true" fixLnInst="true"/>
   </Services>
   <AccessPoint name="S1">
      <Server>
         <Authentication none="true"/>
         <LDevice inst="LD1">
            <LN0 lnClass="LLN0" inst="" lnType="LN0">
               <DataSet name="EventsMmxuMaxAPhs" desc="EventsMmxuMaxAPhs">
                  <FCDA ldInst="LD1" lnClass="MMXU" lnInst="1" doName="MaxAPhs" fc="MX"/>
               </DataSet>
               <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                  <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Str" fc="ST"/>
               </DataSet>
```

```
                    <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                       <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Op" fc="ST"/>
                    </DataSet>
                    <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                       <FCDA ldInst="LD1" lnClass="XCBR" lnInst="1" doName="Pos" fc="ST"/>
                    </DataSet>
                    <ReportControl rptID="EventsMmxuMaxAPhs" confRev="1" buffered="false" bufTime="50
                         " intgPd="1000" name="EventsMmxuMaxAPhsRCB" datSet="EventsMmxuMaxAPhs">
                       <TrgOps period="true"/>
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            dataRef="true" configRef="true"/>
                       <RptEnabled max="1"/>
                    </ReportControl>
                    <ReportControl rptID="EventsPiocStr" confRev="1" buffered="false" bufTime="50"
                         intgPd="1000" name="EventsPiocStrRCB" datSet="EventsPiocStr">
                       <TrgOps period="true"/>
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            dataRef="true" configRef="true"/>
                       <RptEnabled max="1"/>
                    </ReportControl>
                    <ReportControl rptID="EventsPiocOp" confRev="1" buffered="false" bufTime="50"
                         intgPd="1000" name="EventsPiocOpRCB" datSet="EventsPiocOp">
                       <TrgOps period="true"/>
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            dataRef="true" configRef="true"/>
                       <RptEnabled max="1"/>
                    </ReportControl>
                    <ReportControl rptID="EventsXcbrPos" confRev="1" buffered="false" bufTime="50"
                         intgPd="1000" name="EventsXcbrPosRCB" datSet="EventsXcbrPos">
                       <TrgOps period="true"/>
                       <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                            dataRef="true" configRef="true"/>
                       <RptEnabled max="1"/>
                    </ReportControl>
                    <GSEControl type="GOOSE" appID="eventsMmxuMaxAPhs" confRev="2" name="
                         gcbEventsMmxuMaxAPhs" datSet="EventsMmxuMaxAPhs"/>
                    <GSEControl type="GOOSE" appID="eventsPiocStr" confRev="2" name="gcbEventsPiocStr
                         " datSet="EventsPiocStr"/>
                    <GSEControl type="GOOSE" appID="eventsPiocOp" confRev="2" name="gcbEventsPiocOp"
                         datSet="EventsPiocOp"/>
                    <GSEControl type="GOOSE" appID="eventsXcbrPos" confRev="2" name="gcbEventsXcbrPos
                         " datSet="EventsXcbrPos"/>
                 </LN0>
                 <LN lnClass="MMXU" inst="1" lnType="MMXUa"/>
                 <LN lnClass="PIOC" inst="1" lnType="PIOCa">
                    <Inputs>
                       <ExtRef iedName="SSLQ7IED1" ldInst="LD1" lnClass="MMXU" lnInst="1" doName="
                            MaxAPhs" srcLNClass="LLN0" srcCBName="gcbEventsMmxuMaxAPhs"/>
                    </Inputs>
                 </LN>
                 <LN lnClass="XCBR" inst="1" lnType="XCBRa"/>
              </LDevice>
           </Server>
        </AccessPoint>
     </IED>
     <DataTypeTemplates>
        <LNodeType lnClass="LLN0" id="LN0">
           <DO name="Mod" type="myMod"/>
           <DO name="Beh" type="myBeh"/>
           <DO name="Health" type="myHealth"/>
           <DO name="NamPlt" type="myLN0LPL"/>
        </LNodeType>
        <LNodeType lnClass="LPHD" id="LPHDa">
           <DO name="PhyNam" type="myDPL"/>
           <DO name="PhyHealth" type="myINS"/>
           <DO name="Proxy" type="mySPS"/>
        </LNodeType>
        <LNodeType lnClass="XCBR" id="XCBRa">
           <DO name="Beh" type="myBeh"/>
           <DO name="LOC" type="mySPS"/>
           <DO name="OpCnt" type="myINS"/>
           <DO name="Pos" type="myPos"/>
           <DO name="BlkOpn" type="mySPC"/>
```

```
    <DO name="BlkCls" type="mySPC"/>
  </LNodeType>
  <LNodeType lnClass="MMXU" id="MMXUa">
    <DO name="Beh" type="myBeh"/>
    <DO name="TotW" type="myMV"/>
    <DO name="TotVAr" type="myMV"/>
    <DO name="TotVA" type="myMV"/>
    <DO name="Hz" type="myMV"/>
    <DO name="MaxAPhs" type="myMV"/>
  </LNodeType>
  <LNodeType lnClass="PIOC" id="PIOCa">
    <DO name="Beh" type="myBeh"/>
    <DO name="Str" type="myACD"/>
    <DO name="Op" type="myACT"/>
    <DO name="StrVal" type="myASG"/>
  </LNodeType>
  <LNodeType lnClass="PTRC" id="PTRCa">
    <DO name="Beh" type="myBeh"/>
    <DO name="Str" type="myACD"/>
    <DO name="Op" type="myACT"/>
    <DO name="Tr" type="myACT"/>
  </LNodeType>
  <DOType cdc="INC" id="myMod">
    <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Mod"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
    <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
    <DA fc="CO" name="Oper" bType="Struct" type="myModOper"/>
  </DOType>
  <DOType cdc="INS" id="myBeh">
    <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Beh"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
  </DOType>
  <DOType cdc="INS" id="myINS">
    <DA fc="ST" dchg="true" name="stVal" bType="INT32"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
  </DOType>
  <DOType cdc="INS" id="myHealth">
    <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Health"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
  </DOType>
  <DOType cdc="LPL" id="myLN0LPL">
    <DA fc="DC" name="vendor" bType="VisString255"/>
    <DA fc="DC" name="swRev" bType="VisString255"/>
    <DA fc="DC" name="d" bType="VisString255"/>
    <DA fc="DC" name="configRev" bType="VisString255"/>
    <DA fc="EX" name="ldNs" bType="VisString255"/>
  </DOType>
  <DOType cdc="DPC" id="myPos">
    <DA fc="ST" dchg="true" name="stVal" bType="Dbpos"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
    <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
    <DA fc="CO" name="Oper" bType="Struct" type="myOper"/>
    <DA fc="CO" name="SBOw" bType="Struct" type="myOper"/>
    <DA fc="CO" name="Cancel" bType="Struct" type="myCancel"/>
  </DOType>
  <DOType cdc="SPS" id="mySPS">
    <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
    <DA fc="ST" qchg="true" name="q" bType="Quality"/>
    <DA fc="ST" name="t" bType="Timestamp"/>
  </DOType>
  <DOType cdc="MV" id="myMV">
    <DA fc="MX" dchg="true" name="mag" bType="Struct" type="myAnalogValue"/>
    <DA fc="MX" qchg="true" name="q" bType="Quality"/>
    <DA fc="MX" name="t" bType="Timestamp"/>
    <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
  </DOType>
  <DOType cdc="ACD" id="myACD">
    <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
```

```
      <DA fc="ST" dchg="true" name="dirGeneral" bType="Enum" type="ACDdir"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
   </DOType>
   <DOType cdc="ACT" id="myACT">
      <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
   </DOType>
   <DOType cdc="ASG" id="myASG">
      <DA fc="SP" dchg="true" name="setMag" bType="Struct" type="myAnalogValue"/>
      <DA fc="CF" dchg="true" name="units" bType="Enum" type="SIUnit"/>
      <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
   </DOType>
   <DOType cdc="SPC" id="mySPC">
      <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
      <DA fc="ST" qchg="true" name="q" bType="Quality"/>
      <DA fc="ST" name="t" bType="Timestamp"/>
   </DOType>
   <DOType cdc="DPL" id="myDPL">
      <DA fc="DC" name="vendor" bType="VisString255"/>
      <DA fc="DC" name="hwRev" bType="VisString255"/>
   </DOType>
   <DAType id="myAnalogValue">
      <BDA name="f" bType="FLOAT32"/>
   </DAType>
   <DAType id="ScaledValueConfig">
      <BDA name="scaleFactor" bType="FLOAT32"/>
      <BDA name="offset" bType="FLOAT32"/>
   </DAType>
   <DAType id="myModOper">
      <BDA name="ctlVal" bType="Enum" type="Mod"/>
      <BDA name="origin" bType="Struct" type="originator"/>
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <BDA name="Check" bType="Check"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="myOper">
      <BDA name="ctlVal" bType="BOOLEAN"/>
      <BDA name="origin" bType="Struct" type="originator"/>
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <BDA name="Check" bType="Check"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="myCancel">
      <BDA name="ctlVal" bType="BOOLEAN"/>
      <BDA name="origin" bType="Struct" type="originator"/>
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="originator">
      <BDA name="orCat" bType="Enum" type="orCategory" cond="M"/>
      <BDA name="orIdent" bType="Octet64" cond="M"/>
   </DAType>
   <EnumType id="ACDdir">
      <EnumVal ord="0">unknown</EnumVal>
      <EnumVal ord="1">forward</EnumVal>
      <EnumVal ord="2">backward</EnumVal>
      <EnumVal ord="3">both</EnumVal>
   </EnumType>
   <EnumType id="Health">
      <EnumVal ord="1">Ok</EnumVal>
      <EnumVal ord="2">Warning</EnumVal>
      <EnumVal ord="3">Alarm</EnumVal>
   </EnumType>
   <EnumType id="ctlModel">
      <EnumVal ord="0">status-only</EnumVal>
```

```
      <EnumVal ord="1">direct-with-normal-security</EnumVal>
      <EnumVal ord="2">sbo-with-normal-security</EnumVal>
      <EnumVal ord="3">direct-with-enhanced-security</EnumVal>
      <EnumVal ord="4">sbo-with-enhanced-security</EnumVal>
   </EnumType>
   <EnumType id="orCategory">
      <EnumVal ord="0">not-supported</EnumVal>
      <EnumVal ord="1">bay-control</EnumVal>
      <EnumVal ord="2">station-control</EnumVal>
      <EnumVal ord="3">remote-control</EnumVal>
      <EnumVal ord="4">automatic-bay</EnumVal>
      <EnumVal ord="5">automatic-station</EnumVal>
      <EnumVal ord="6">automatic-remote</EnumVal>
      <EnumVal ord="7">maintenance</EnumVal>
      <EnumVal ord="8">process</EnumVal>
   </EnumType>
   <EnumType id="Beh">
      <EnumVal ord="1">on</EnumVal>
      <EnumVal ord="2">blocked</EnumVal>
      <EnumVal ord="3">test</EnumVal>
      <EnumVal ord="4">test/blocked</EnumVal>
      <EnumVal ord="5">off</EnumVal>
   </EnumType>
   <EnumType id="Mod">
      <EnumVal ord="1">on</EnumVal>
      <EnumVal ord="2">blocked</EnumVal>
      <EnumVal ord="3">test</EnumVal>
      <EnumVal ord="4">test/blocked</EnumVal>
      <EnumVal ord="5">off</EnumVal>
   </EnumType>
   <EnumType id="Health">
      <EnumVal ord="1">Ok</EnumVal>
      <EnumVal ord="2">Warning</EnumVal>
      <EnumVal ord="3">Alarm</EnumVal>
   </EnumType>
   <EnumType id="SIUnit">
      <EnumVal ord="1"></EnumVal>
      <EnumVal ord="2">m</EnumVal>
      <EnumVal ord="3">kg</EnumVal>
      <EnumVal ord="4">s</EnumVal>
      <EnumVal ord="5">A</EnumVal>
      <EnumVal ord="6">K</EnumVal>
      <EnumVal ord="7">mol</EnumVal>
      <EnumVal ord="8">cd</EnumVal>
      <EnumVal ord="9">deg</EnumVal>
      <EnumVal ord="10">rad</EnumVal>
      <EnumVal ord="11">sr</EnumVal>
      <EnumVal ord="21">Gy</EnumVal>
      <EnumVal ord="22">q</EnumVal>
      <EnumVal ord="23">°C</EnumVal>
      <EnumVal ord="24">Sv</EnumVal>
      <EnumVal ord="25">F</EnumVal>
      <EnumVal ord="26">C</EnumVal>
      <EnumVal ord="27">S</EnumVal>
      <EnumVal ord="28">H</EnumVal>
      <EnumVal ord="29">V</EnumVal>
      <EnumVal ord="30">ohm</EnumVal>
      <EnumVal ord="31">J</EnumVal>
      <EnumVal ord="32">N</EnumVal>
      <EnumVal ord="33">Hz</EnumVal>
      <EnumVal ord="34">lx</EnumVal>
      <EnumVal ord="35">Lm</EnumVal>
      <EnumVal ord="36">Wb</EnumVal>
      <EnumVal ord="37">T</EnumVal>
      <EnumVal ord="38">W</EnumVal>
      <EnumVal ord="39">Pa</EnumVal>
      <EnumVal ord="41">m2</EnumVal>
      <EnumVal ord="42">m3</EnumVal>
      <EnumVal ord="43">m/s</EnumVal>
      <EnumVal ord="44">m/s2</EnumVal>
      <EnumVal ord="45">m3/s</EnumVal>
      <EnumVal ord="46">m/m3</EnumVal>
      <EnumVal ord="47">M</EnumVal>
```

```
            <EnumVal ord="48">kg/m3</EnumVal>
            <EnumVal ord="49">m2/s</EnumVal>
            <EnumVal ord="50">W/m K</EnumVal>
            <EnumVal ord="51">J/K</EnumVal>
            <EnumVal ord="52">ppm</EnumVal>
            <EnumVal ord="53">1/s</EnumVal>
            <EnumVal ord="54">rad/s</EnumVal>
            <EnumVal ord="61">VA</EnumVal>
            <EnumVal ord="62">Watts</EnumVal>
            <EnumVal ord="63">VAr</EnumVal>
            <EnumVal ord="64">phi</EnumVal>
            <EnumVal ord="65">cos(phi)</EnumVal>
            <EnumVal ord="66">Vs</EnumVal>
            <EnumVal ord="67">V2</EnumVal>
            <EnumVal ord="68">As</EnumVal>
            <EnumVal ord="69">A2</EnumVal>
            <EnumVal ord="70">A2t</EnumVal>
            <EnumVal ord="71">VAh</EnumVal>
            <EnumVal ord="72">Wh</EnumVal>
            <EnumVal ord="73">VArh</EnumVal>
            <EnumVal ord="74">V/Hz</EnumVal>
            <EnumVal ord="75">Hz/s</EnumVal>
            <EnumVal ord="76">char</EnumVal>
            <EnumVal ord="77">char/s</EnumVal>
            <EnumVal ord="78">kgm2</EnumVal>
            <EnumVal ord="79">dB</EnumVal>
         </EnumType>
      </DataTypeTemplates>
</SCL>
```

Listing 7.72: Configured IED Description File for IED 2

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SCL xmlns="http://www.iec.ch/61850/2003/SCL" version="2007" revision="A">
   <Header id="SSD Example"/>
   <Communication>
      <SubNetwork type="IP" name="Mgmt" desc="">
         <Text>Management bus</Text>
      </SubNetwork>
      <SubNetwork type="8-MMS" name="Station" desc="">
         <Text>Station bus</Text>
         <BitRate unit="b/s">10</BitRate>
         <ConnectedAP iedName="SSLQ7IED3" apName="S1">
            <Address>
               <P type="IP">10.0.0.133</P>
               <P type="IP-SUBNET">255.255.255.0</P>
               <P type="MAC-Address">02-42-c0-a8-84-23</P>
            </Address>
            <GSE ldInst="LD1" cbName="gcbEventsPtrcTr">
               <Address>
                  <P type="MAC-Address">02-42-c0-a8-84-21</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
            <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
               <Address>
                  <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
            <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
               <Address>
                  <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
```

```
            <GSE ldInst="LD1" cbName="gcbEventsXcbrPos">
               <Address>
                  <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
         </ConnectedAP>
      </SubNetwork>
   </Communication>
   <IED name="SSLQ7IED3" type="SN0003" manufacturer="Siemens">
      <Private type="Programmable">true</Private>
      <Private type="Memory">10000</Private>
      <Services>
         <DynAssociation/>
         <GetDirectory/>
         <GetDataObjectDefinition/>
         <GetDataSetValue/>
         <DataSetDirectory/>
         <ReadWrite/>
         <TimerActivatedControl/>
         <GetCBValues/>
         <GSEDir/>
         <GOOSE max="5"/>
         <GSSE max="5"/>
         <FileHandling/>
         <ConfLNs fixPrefix="true" fixLnInst="true"/>
      </Services>
      <AccessPoint name="S1">
         <Server>
            <Authentication none="true"/>
            <LDevice inst="LD1">
               <LN0 lnClass="LLN0" inst="" lnType="LN0">
                  <DataSet name="EventsPtrcTr" desc="EventsPtrcTr">
                     <FCDA ldInst="LD1" lnClass="PTRC" lnInst="1" doName="Tr" fc="ST"/>
                  </DataSet>
                  <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                     <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Str" fc="ST"/>
                  </DataSet>
                  <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                     <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Op" fc="ST"/>
                  </DataSet>
                  <DataSet name="EventsXcbrPos" desc="EventsXcbrPos">
                     <FCDA ldInst="LD1" lnClass="XCBR" lnInst="1" doName="Pos" fc="ST"/>
                  </DataSet>
                  <ReportControl rptID="EventsPtrcTr" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsPtrcTrRCB" datSet="EventsPtrcTr">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                          dataRef="true" configRef="true"/>
                     <RptEnabled max="1"/>
                  </ReportControl>
                  <ReportControl rptID="EventsPiocStr" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsPiocStrRCB" datSet="EventsPiocStr">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                          dataRef="true" configRef="true"/>
                     <RptEnabled max="1"/>
                  </ReportControl>
                  <ReportControl rptID="EventsPiocOp" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsPiocOpRCB" datSet="EventsPiocOp">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                          dataRef="true" configRef="true"/>
                     <RptEnabled max="1"/>
                  </ReportControl>
                  <ReportControl rptID="EventsXcbrPos" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsXcbrPosRCB" datSet="EventsXcbrPos">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                          dataRef="true" configRef="true"/>
                     <RptEnabled max="1"/>
```

```
                    </ReportControl>
                    <GSEControl type="GOOSE" appID="eventsPtrcTr" confRev="2" name="gcbEventsPtrcTr"
                        datSet="EventsPtrcTr"/>
                    <GSEControl type="GOOSE" appID="eventsPiocStr" confRev="2" name="gcbEventsPiocStr
                        " datSet="EventsPiocStr"/>
                    <GSEControl type="GOOSE" appID="eventsPiocOp" confRev="2" name="gcbEventsPiocOp"
                        datSet="EventsPiocOp"/>
                    <GSEControl type="GOOSE" appID="eventsXcbrPos" confRev="2" name="gcbEventsXcbrPos
                        " datSet="EventsXcbrPos"/>
                </LN0>
                <LN lnClass="PTRC" inst="1" lnType="PTRCa">
                    <Inputs>
                        <ExtRef iedName="SSLQ7IED2" ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Op"
                            srcLNClass="LLN0" srcCBName="gcbEventsPiocOp"/>
                    </Inputs>
                </LN>
                <LN lnClass="PIOC" inst="1" lnType="PIOCa"/>
                <LN lnClass="XCBR" inst="1" lnType="XCBRa"/>
            </LDevice>
        </Server>
    </AccessPoint>
</IED>
<DataTypeTemplates>
    <LNodeType lnClass="LLN0" id="LN0">
        <DO name="Mod" type="myMod"/>
        <DO name="Beh" type="myBeh"/>
        <DO name="Health" type="myHealth"/>
        <DO name="NamPlt" type="myLN0LPL"/>
    </LNodeType>
    <LNodeType lnClass="LPHD" id="LPHDa">
        <DO name="PhyNam" type="myDPL"/>
        <DO name="PhyHealth" type="myINS"/>
        <DO name="Proxy" type="mySPS"/>
    </LNodeType>
    <LNodeType lnClass="XCBR" id="XCBRa">
        <DO name="Beh" type="myBeh"/>
        <DO name="LOC" type="mySPS"/>
        <DO name="OpCnt" type="myINS"/>
        <DO name="Pos" type="myPos"/>
        <DO name="BlkOpn" type="mySPC"/>
        <DO name="BlkCls" type="mySPC"/>
    </LNodeType>
    <LNodeType lnClass="MMXU" id="MMXUa">
        <DO name="Beh" type="myBeh"/>
        <DO name="TotW" type="myMV"/>
        <DO name="TotVAr" type="myMV"/>
        <DO name="TotVA" type="myMV"/>
        <DO name="Hz" type="myMV"/>
        <DO name="MaxAPhs" type="myMV"/>
    </LNodeType>
    <LNodeType lnClass="PIOC" id="PIOCa">
        <DO name="Beh" type="myBeh"/>
        <DO name="Str" type="myACD"/>
        <DO name="Op" type="myACT"/>
        <DO name="StrVal" type="myASG"/>
    </LNodeType>
    <LNodeType lnClass="PTRC" id="PTRCa">
        <DO name="Beh" type="myBeh"/>
        <DO name="Str" type="myACD"/>
        <DO name="Op" type="myACT"/>
        <DO name="Tr" type="myACT"/>
    </LNodeType>
    <DOType cdc="INC" id="myMod">
        <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Mod"/>
        <DA fc="ST" qchg="true" name="q" bType="Quality"/>
        <DA fc="ST" name="t" bType="Timestamp"/>
        <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
        <DA fc="CO" name="Oper" bType="Struct" type="myModOper"/>
    </DOType>
    <DOType cdc="INS" id="myBeh">
        <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Beh"/>
        <DA fc="ST" qchg="true" name="q" bType="Quality"/>
        <DA fc="ST" name="t" bType="Timestamp"/>
```

```
</DOType>
<DOType cdc="INS" id="myINS">
   <DA fc="ST" dchg="true" name="stVal" bType="INT32"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="INS" id="myHealth">
   <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Health"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="LPL" id="myLN0LPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="swRev" bType="VisString255"/>
   <DA fc="DC" name="d" bType="VisString255"/>
   <DA fc="DC" name="configRev" bType="VisString255"/>
   <DA fc="EX" name="ldNs" bType="VisString255"/>
</DOType>
<DOType cdc="DPC" id="myPos">
   <DA fc="ST" dchg="true" name="stVal" bType="Dbpos"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
   <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
   <DA fc="CO" name="Oper" bType="Struct" type="myOper"/>
   <DA fc="CO" name="SBOw" bType="Struct" type="myOper"/>
   <DA fc="CO" name="Cancel" bType="Struct" type="myCancel"/>
</DOType>
<DOType cdc="SPS" id="mySPS">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="MV" id="myMV">
   <DA fc="MX" dchg="true" name="mag" bType="Struct" type="myAnalogValue"/>
   <DA fc="MX" qchg="true" name="q" bType="Quality"/>
   <DA fc="MX" name="t" bType="Timestamp"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="ACD" id="myACD">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" dchg="true" name="dirGeneral" bType="Enum" type="ACDdir"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ACT" id="myACT">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ASG" id="myASG">
   <DA fc="SP" dchg="true" name="setMag" bType="Struct" type="myAnalogValue"/>
   <DA fc="CF" dchg="true" name="units" bType="Enum" type="SIUnit"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="SPC" id="mySPC">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="DPL" id="myDPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="hwRev" bType="VisString255"/>
</DOType>
<DAType id="myAnalogValue">
   <BDA name="f" bType="FLOAT32"/>
</DAType>
<DAType id="ScaledValueConfig">
   <BDA name="scaleFactor" bType="FLOAT32"/>
   <BDA name="offset" bType="FLOAT32"/>
</DAType>
<DAType id="myModOper">
   <BDA name="ctlVal" bType="Enum" type="Mod"/>
   <BDA name="origin" bType="Struct" type="originator"/>
```

```
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <BDA name="Check" bType="Check"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="myOper">
      <BDA name="ctlVal" bType="BOOLEAN"/>
      <BDA name="origin" bType="Struct" type="originator"/>
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <BDA name="Check" bType="Check"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="myCancel">
      <BDA name="ctlVal" bType="BOOLEAN"/>
      <BDA name="origin" bType="Struct" type="originator"/>
      <BDA name="ctlNum" bType="INT8U"/>
      <BDA name="T" bType="Timestamp"/>
      <BDA name="Test" bType="BOOLEAN"/>
      <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
   </DAType>
   <DAType id="originator">
      <BDA name="orCat" bType="Enum" type="orCategory" cond="M"/>
      <BDA name="orIdent" bType="Octet64" cond="M"/>
   </DAType>
   <EnumType id="ACDdir">
      <EnumVal ord="0">unknown</EnumVal>
      <EnumVal ord="1">forward</EnumVal>
      <EnumVal ord="2">backward</EnumVal>
      <EnumVal ord="3">both</EnumVal>
   </EnumType>
   <EnumType id="Health">
      <EnumVal ord="1">Ok</EnumVal>
      <EnumVal ord="2">Warning</EnumVal>
      <EnumVal ord="3">Alarm</EnumVal>
   </EnumType>
   <EnumType id="ctlModel">
      <EnumVal ord="0">status-only</EnumVal>
      <EnumVal ord="1">direct-with-normal-security</EnumVal>
      <EnumVal ord="2">sbo-with-normal-security</EnumVal>
      <EnumVal ord="3">direct-with-enhanced-security</EnumVal>
      <EnumVal ord="4">sbo-with-enhanced-security</EnumVal>
   </EnumType>
   <EnumType id="orCategory">
      <EnumVal ord="0">not-supported</EnumVal>
      <EnumVal ord="1">bay-control</EnumVal>
      <EnumVal ord="2">station-control</EnumVal>
      <EnumVal ord="3">remote-control</EnumVal>
      <EnumVal ord="4">automatic-bay</EnumVal>
      <EnumVal ord="5">automatic-station</EnumVal>
      <EnumVal ord="6">automatic-remote</EnumVal>
      <EnumVal ord="7">maintenance</EnumVal>
      <EnumVal ord="8">process</EnumVal>
   </EnumType>
   <EnumType id="Beh">
      <EnumVal ord="1">on</EnumVal>
      <EnumVal ord="2">blocked</EnumVal>
      <EnumVal ord="3">test</EnumVal>
      <EnumVal ord="4">test/blocked</EnumVal>
      <EnumVal ord="5">off</EnumVal>
   </EnumType>
   <EnumType id="Mod">
      <EnumVal ord="1">on</EnumVal>
      <EnumVal ord="2">blocked</EnumVal>
      <EnumVal ord="3">test</EnumVal>
      <EnumVal ord="4">test/blocked</EnumVal>
      <EnumVal ord="5">off</EnumVal>
   </EnumType>
   <EnumType id="Health">
      <EnumVal ord="1">Ok</EnumVal>
      <EnumVal ord="2">Warning</EnumVal>
```

```
            <EnumVal ord="3">Alarm</EnumVal>
         </EnumType>
         <EnumType id="SIUnit">
            <EnumVal ord="1"></EnumVal>
            <EnumVal ord="2">m</EnumVal>
            <EnumVal ord="3">kg</EnumVal>
            <EnumVal ord="4">s</EnumVal>
            <EnumVal ord="5">A</EnumVal>
            <EnumVal ord="6">K</EnumVal>
            <EnumVal ord="7">mol</EnumVal>
            <EnumVal ord="8">cd</EnumVal>
            <EnumVal ord="9">deg</EnumVal>
            <EnumVal ord="10">rad</EnumVal>
            <EnumVal ord="11">sr</EnumVal>
            <EnumVal ord="21">Gy</EnumVal>
            <EnumVal ord="22">q</EnumVal>
            <EnumVal ord="23">°C</EnumVal>
            <EnumVal ord="24">Sv</EnumVal>
            <EnumVal ord="25">F</EnumVal>
            <EnumVal ord="26">C</EnumVal>
            <EnumVal ord="27">S</EnumVal>
            <EnumVal ord="28">H</EnumVal>
            <EnumVal ord="29">V</EnumVal>
            <EnumVal ord="30">ohm</EnumVal>
            <EnumVal ord="31">J</EnumVal>
            <EnumVal ord="32">N</EnumVal>
            <EnumVal ord="33">Hz</EnumVal>
            <EnumVal ord="34">lx</EnumVal>
            <EnumVal ord="35">Lm</EnumVal>
            <EnumVal ord="36">Wb</EnumVal>
            <EnumVal ord="37">T</EnumVal>
            <EnumVal ord="38">W</EnumVal>
            <EnumVal ord="39">Pa</EnumVal>
            <EnumVal ord="41">m2</EnumVal>
            <EnumVal ord="42">m3</EnumVal>
            <EnumVal ord="43">m/s</EnumVal>
            <EnumVal ord="44">m/s2</EnumVal>
            <EnumVal ord="45">m3/s</EnumVal>
            <EnumVal ord="46">m/m3</EnumVal>
            <EnumVal ord="47">M</EnumVal>
            <EnumVal ord="48">kg/m3</EnumVal>
            <EnumVal ord="49">m2/s</EnumVal>
            <EnumVal ord="50">W/m K</EnumVal>
            <EnumVal ord="51">J/K</EnumVal>
            <EnumVal ord="52">ppm</EnumVal>
            <EnumVal ord="53">1/s</EnumVal>
            <EnumVal ord="54">rad/s</EnumVal>
            <EnumVal ord="61">VA</EnumVal>
            <EnumVal ord="62">Watts</EnumVal>
            <EnumVal ord="63">VAr</EnumVal>
            <EnumVal ord="64">phi</EnumVal>
            <EnumVal ord="65">cos(phi)</EnumVal>
            <EnumVal ord="66">Vs</EnumVal>
            <EnumVal ord="67">V2</EnumVal>
            <EnumVal ord="68">As</EnumVal>
            <EnumVal ord="69">A2</EnumVal>
            <EnumVal ord="70">A2t</EnumVal>
            <EnumVal ord="71">VAh</EnumVal>
            <EnumVal ord="72">Wh</EnumVal>
            <EnumVal ord="73">VArh</EnumVal>
            <EnumVal ord="74">V/Hz</EnumVal>
            <EnumVal ord="75">Hz/s</EnumVal>
            <EnumVal ord="76">char</EnumVal>
            <EnumVal ord="77">char/s</EnumVal>
            <EnumVal ord="78">kgm2</EnumVal>
            <EnumVal ord="79">dB</EnumVal>
         </EnumType>
      </DataTypeTemplates>
</SCL>
```

Listing 7.73: Configured IED Description File for IED 3

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

```
<SCL xmlns="http://www.iec.ch/61850/2003/SCL" version="2007" revision="A">
   <Header id="SSD Example"/>
   <Communication>
      <SubNetwork type="IP" name="Mgmt" desc="">
         <Text>Management bus</Text>
      </SubNetwork>
      <SubNetwork type="8-MMS" name="Station" desc="">
         <Text>Station bus</Text>
         <BitRate unit="b/s">10</BitRate>
         <ConnectedAP iedName="SSLQ7IED4" apName="S1">
            <Address>
               <P type="IP">10.0.0.134</P>
               <P type="IP-SUBNET">255.255.255.0</P>
               <P type="MAC-Address">02-42-c0-a8-84-24</P>
            </Address>
            <GSE ldInst="LD1" cbName="gcbEventsPiocStr">
               <Address>
                  <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
            <GSE ldInst="LD1" cbName="gcbEventsPiocOp">
               <Address>
                  <P type="MAC-Address">ff-ff-ff-ff-ff-ff</P>
                  <P type="APPID">1000</P>
               </Address>
               <MinTime>1000</MinTime>
               <MaxTime>3000</MaxTime>
            </GSE>
         </ConnectedAP>
      </SubNetwork>
   </Communication>
   <IED name="SSLQ7IED4" type="SN0004" manufacturer="Schneider">
      <Private type="Programmable">true</Private>
      <Private type="Memory">10000</Private>
      <Services>
         <DynAssociation/>
         <GetDirectory/>
         <GetDataObjectDefinition/>
         <GetDataSetValue/>
         <DataSetDirectory/>
         <ReadWrite/>
         <TimerActivatedControl/>
         <GetCBValues/>
         <GSEDir/>
         <GOOSE max="5"/>
         <GSSE max="5"/>
         <FileHandling/>
         <ConfLNs fixPrefix="true" fixLnInst="true"/>
      </Services>
      <AccessPoint name="S1" router="true">
         <Server>
            <LDevice inst="LD1">
               <LN0 lnClass="LLN0" inst="" lnType="LN0">
                  <DataSet name="EventsPiocStr" desc="EventsPiocStr">
                     <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Str" fc="ST"/>
                  </DataSet>
                  <DataSet name="EventsPiocOp" desc="EventsPiocOp">
                     <FCDA ldInst="LD1" lnClass="PIOC" lnInst="1" doName="Op" fc="ST"/>
                  </DataSet>
                  <ReportControl rptID="EventsPiocStr" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsPiocStrRCB" datSet="EventsPiocStr">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
                          dataRef="true" configRef="true"/>
                     <RptEnabled max="1"/>
                  </ReportControl>
                  <ReportControl rptID="EventsPiocOp" confRev="1" buffered="false" bufTime="50"
                       intgPd="1000" name="EventsPiocOpRCB" datSet="EventsPiocOp">
                     <TrgOps period="true"/>
                     <OptFields seqNum="true" timeStamp="true" dataSet="true" reasonCode="true"
```

```
                        dataRef="true" configRef="true"/>
                    <RptEnabled max="1"/>
                </ReportControl>
                <GSEControl type="GOOSE" appID="eventsPiocStr" confRev="2" name="gcbEventsPiocStr
                    " datSet="EventsPiocStr"/>
                <GSEControl type="GOOSE" appID="eventsPiocOp" confRev="2" name="gcbEventsPiocOp"
                    datSet="EventsPiocOp"/>
            </LN0>
            <LN lnClass="PIOC" inst="1" lnType="PIOCa"/>
        </LDevice>
    </Server>
</AccessPoint>
<AccessPoint name="S2">
    <ServerAt apName="S2"/>
</AccessPoint>
</IED>
<DataTypeTemplates>
    <LNodeType lnClass="LLN0" id="LN0">
        <DO name="Mod" type="myMod"/>
        <DO name="Beh" type="myBeh"/>
        <DO name="Health" type="myHealth"/>
        <DO name="NamPlt" type="myLN0LPL"/>
    </LNodeType>
    <LNodeType lnClass="LPHD" id="LPHDa">
        <DO name="PhyNam" type="myDPL"/>
        <DO name="PhyHealth" type="myINS"/>
        <DO name="Proxy" type="mySPS"/>
    </LNodeType>
    <LNodeType lnClass="XCBR" id="XCBRa">
        <DO name="Beh" type="myBeh"/>
        <DO name="LOC" type="mySPS"/>
        <DO name="OpCnt" type="myINS"/>
        <DO name="Pos" type="myPos"/>
        <DO name="BlkOpn" type="mySPC"/>
        <DO name="BlkCls" type="mySPC"/>
    </LNodeType>
    <LNodeType lnClass="MMXU" id="MMXUa">
        <DO name="Beh" type="myBeh"/>
        <DO name="TotW" type="myMV"/>
        <DO name="TotVAr" type="myMV"/>
        <DO name="TotVA" type="myMV"/>
        <DO name="Hz" type="myMV"/>
        <DO name="MaxAPhs" type="myMV"/>
    </LNodeType>
    <LNodeType lnClass="PIOC" id="PIOCa">
        <DO name="Beh" type="myBeh"/>
        <DO name="Str" type="myACD"/>
        <DO name="Op" type="myACT"/>
        <DO name="StrVal" type="myASG"/>
    </LNodeType>
    <LNodeType lnClass="PTRC" id="PTRCa">
        <DO name="Beh" type="myBeh"/>
        <DO name="Str" type="myACD"/>
        <DO name="Op" type="myACT"/>
        <DO name="Tr" type="myACT"/>
    </LNodeType>
    <DOType cdc="INC" id="myMod">
        <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Mod"/>
        <DA fc="ST" qchg="true" name="q" bType="Quality"/>
        <DA fc="ST" name="t" bType="Timestamp"/>
        <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
        <DA fc="CO" name="Oper" bType="Struct" type="myModOper"/>
    </DOType>
    <DOType cdc="INS" id="myBeh">
        <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Beh"/>
        <DA fc="ST" qchg="true" name="q" bType="Quality"/>
        <DA fc="ST" name="t" bType="Timestamp"/>
    </DOType>
    <DOType cdc="INS" id="myINS">
        <DA fc="ST" dchg="true" name="stVal" bType="INT32"/>
        <DA fc="ST" qchg="true" name="q" bType="Quality"/>
        <DA fc="ST" name="t" bType="Timestamp"/>
    </DOType>
```

```
<DOType cdc="INS" id="myHealth">
   <DA fc="ST" dchg="true" name="stVal" bType="Enum" type="Health"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="LPL" id="myLN0LPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="swRev" bType="VisString255"/>
   <DA fc="DC" name="d" bType="VisString255"/>
   <DA fc="DC" name="configRev" bType="VisString255"/>
   <DA fc="EX" name="ldNs" bType="VisString255"/>
</DOType>
<DOType cdc="DPC" id="myPos">
   <DA fc="ST" dchg="true" name="stVal" bType="Dbpos"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
   <DA fc="CF" name="ctlModel" bType="Enum" type="ctlModel"/>
   <DA fc="CO" name="Oper" bType="Struct" type="myOper"/>
   <DA fc="CO" name="SBOw" bType="Struct" type="myOper"/>
   <DA fc="CO" name="Cancel" bType="Struct" type="myCancel"/>
</DOType>
<DOType cdc="SPS" id="mySPS">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="MV" id="myMV">
   <DA fc="MX" dchg="true" name="mag" bType="Struct" type="myAnalogValue"/>
   <DA fc="MX" qchg="true" name="q" bType="Quality"/>
   <DA fc="MX" name="t" bType="Timestamp"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="ACD" id="myACD">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" dchg="true" name="dirGeneral" bType="Enum" type="ACDdir"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ACT" id="myACT">
   <DA fc="ST" dchg="true" name="general" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="ASG" id="myASG">
   <DA fc="SP" dchg="true" name="setMag" bType="Struct" type="myAnalogValue"/>
   <DA fc="CF" dchg="true" name="units" bType="Enum" type="SIUnit"/>
   <DA fc="CF" dchg="true" name="sVC" bType="Struct" type="ScaledValueConfig"/>
</DOType>
<DOType cdc="SPC" id="mySPC">
   <DA fc="ST" dchg="true" name="stVal" bType="BOOLEAN"/>
   <DA fc="ST" qchg="true" name="q" bType="Quality"/>
   <DA fc="ST" name="t" bType="Timestamp"/>
</DOType>
<DOType cdc="DPL" id="myDPL">
   <DA fc="DC" name="vendor" bType="VisString255"/>
   <DA fc="DC" name="hwRev" bType="VisString255"/>
</DOType>
<DAType id="myAnalogValue">
   <BDA name="f" bType="FLOAT32"/>
</DAType>
<DAType id="ScaledValueConfig">
   <BDA name="scaleFactor" bType="FLOAT32"/>
   <BDA name="offset" bType="FLOAT32"/>
</DAType>
<DAType id="myModOper">
   <BDA name="ctlVal" bType="Enum" type="Mod"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <BDA name="Check" bType="Check"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
```

```
<DAType id="myOper">
   <BDA name="ctlVal" bType="BOOLEAN"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <BDA name="Check" bType="Check"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="myCancel">
   <BDA name="ctlVal" bType="BOOLEAN"/>
   <BDA name="origin" bType="Struct" type="originator"/>
   <BDA name="ctlNum" bType="INT8U"/>
   <BDA name="T" bType="Timestamp"/>
   <BDA name="Test" bType="BOOLEAN"/>
   <ProtNs type="8-MMS">IEC 61850-8-1:2003</ProtNs>
</DAType>
<DAType id="originator">
   <BDA name="orCat" bType="Enum" type="orCategory" cond="M"/>
   <BDA name="orIdent" bType="Octet64" cond="M"/>
</DAType>
<EnumType id="ACDdir">
   <EnumVal ord="0">unknown</EnumVal>
   <EnumVal ord="1">forward</EnumVal>
   <EnumVal ord="2">backward</EnumVal>
   <EnumVal ord="3">both</EnumVal>
</EnumType>
<EnumType id="Health">
   <EnumVal ord="1">Ok</EnumVal>
   <EnumVal ord="2">Warning</EnumVal>
   <EnumVal ord="3">Alarm</EnumVal>
</EnumType>
<EnumType id="ctlModel">
   <EnumVal ord="0">status-only</EnumVal>
   <EnumVal ord="1">direct-with-normal-security</EnumVal>
   <EnumVal ord="2">sbo-with-normal-security</EnumVal>
   <EnumVal ord="3">direct-with-enhanced-security</EnumVal>
   <EnumVal ord="4">sbo-with-enhanced-security</EnumVal>
</EnumType>
<EnumType id="orCategory">
   <EnumVal ord="0">not-supported</EnumVal>
   <EnumVal ord="1">bay-control</EnumVal>
   <EnumVal ord="2">station-control</EnumVal>
   <EnumVal ord="3">remote-control</EnumVal>
   <EnumVal ord="4">automatic-bay</EnumVal>
   <EnumVal ord="5">automatic-station</EnumVal>
   <EnumVal ord="6">automatic-remote</EnumVal>
   <EnumVal ord="7">maintenance</EnumVal>
   <EnumVal ord="8">process</EnumVal>
</EnumType>
<EnumType id="Beh">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
   <EnumVal ord="4">test/blocked</EnumVal>
   <EnumVal ord="5">off</EnumVal>
</EnumType>
<EnumType id="Mod">
   <EnumVal ord="1">on</EnumVal>
   <EnumVal ord="2">blocked</EnumVal>
   <EnumVal ord="3">test</EnumVal>
   <EnumVal ord="4">test/blocked</EnumVal>
   <EnumVal ord="5">off</EnumVal>
</EnumType>
<EnumType id="Health">
   <EnumVal ord="1">Ok</EnumVal>
   <EnumVal ord="2">Warning</EnumVal>
   <EnumVal ord="3">Alarm</EnumVal>
</EnumType>
<EnumType id="SIUnit">
   <EnumVal ord="1"></EnumVal>
   <EnumVal ord="2">m</EnumVal>
   <EnumVal ord="3">kg</EnumVal>
```

```
            <EnumVal ord="4">s</EnumVal>
            <EnumVal ord="5">A</EnumVal>
            <EnumVal ord="6">K</EnumVal>
            <EnumVal ord="7">mol</EnumVal>
            <EnumVal ord="8">cd</EnumVal>
            <EnumVal ord="9">deg</EnumVal>
            <EnumVal ord="10">rad</EnumVal>
            <EnumVal ord="11">sr</EnumVal>
            <EnumVal ord="21">Gy</EnumVal>
            <EnumVal ord="22">q</EnumVal>
            <EnumVal ord="23">°C</EnumVal>
            <EnumVal ord="24">Sv</EnumVal>
            <EnumVal ord="25">F</EnumVal>
            <EnumVal ord="26">C</EnumVal>
            <EnumVal ord="27">S</EnumVal>
            <EnumVal ord="28">H</EnumVal>
            <EnumVal ord="29">V</EnumVal>
            <EnumVal ord="30">ohm</EnumVal>
            <EnumVal ord="31">J</EnumVal>
            <EnumVal ord="32">N</EnumVal>
            <EnumVal ord="33">Hz</EnumVal>
            <EnumVal ord="34">lx</EnumVal>
            <EnumVal ord="35">Lm</EnumVal>
            <EnumVal ord="36">Wb</EnumVal>
            <EnumVal ord="37">T</EnumVal>
            <EnumVal ord="38">W</EnumVal>
            <EnumVal ord="39">Pa</EnumVal>
            <EnumVal ord="41">m2</EnumVal>
            <EnumVal ord="42">m3</EnumVal>
            <EnumVal ord="43">m/s</EnumVal>
            <EnumVal ord="44">m/s2</EnumVal>
            <EnumVal ord="45">m3/s</EnumVal>
            <EnumVal ord="46">m/m3</EnumVal>
            <EnumVal ord="47">M</EnumVal>
            <EnumVal ord="48">kg/m3</EnumVal>
            <EnumVal ord="49">m2/s</EnumVal>
            <EnumVal ord="50">W/m K</EnumVal>
            <EnumVal ord="51">J/K</EnumVal>
            <EnumVal ord="52">ppm</EnumVal>
            <EnumVal ord="53">1/s</EnumVal>
            <EnumVal ord="54">rad/s</EnumVal>
            <EnumVal ord="61">VA</EnumVal>
            <EnumVal ord="62">Watts</EnumVal>
            <EnumVal ord="63">VAr</EnumVal>
            <EnumVal ord="64">phi</EnumVal>
            <EnumVal ord="65">cos(phi)</EnumVal>
            <EnumVal ord="66">Vs</EnumVal>
            <EnumVal ord="67">V2</EnumVal>
            <EnumVal ord="68">As</EnumVal>
            <EnumVal ord="69">A2</EnumVal>
            <EnumVal ord="70">A2t</EnumVal>
            <EnumVal ord="71">VAh</EnumVal>
            <EnumVal ord="72">Wh</EnumVal>
            <EnumVal ord="73">VArh</EnumVal>
            <EnumVal ord="74">V/Hz</EnumVal>
            <EnumVal ord="75">Hz/s</EnumVal>
            <EnumVal ord="76">char</EnumVal>
            <EnumVal ord="77">char/s</EnumVal>
            <EnumVal ord="78">kgm2</EnumVal>
            <EnumVal ord="79">dB</EnumVal>
        </EnumType>
    </DataTypeTemplates>
</SCL>
```

Listing 7.74: Configured IED Description File for IED 4

7.6.3 System File

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<System Name="diplomarbeit" Comment="">
    <VersionInfo Version="1.0" Author="fxk14i" Date="2021-02-09"/>
    <Application Name="diplomarbeitApp" Comment="">
```

```
<SubAppNetwork>
   <FB Name="iec_61850_MMXU" Type="iec_61850_MMXU" Comment="" x="705.8823529411765" y="
        1647.0588235294117"/>
   <FB Name="iec_61850_PTRC" Type="iec_61850_PTRC" Comment="" x="3764.705882352941" y="
        117.6470588235294"/>
   <FB Name="iec_61850_XCBR" Type="iec_61850_XCBR" Comment="" x="4941.176470588235" y="
        117.6470588235294"/>
   <FB Name="READ_FROM_DB" Type="READ_FROM_DB" Comment="" x="3411.7647058823527" y="
        2470.5882352941176">
      <Parameter Name="ID" Value="10.0.0.130:5000"/>
   </FB>
   <FB Name="STORE_IN_DB" Type="STORE_IN_DB" Comment="" x="6117.647058823529" y="
        1647.0588235294117">
      <Parameter Name="ID" Value="10.0.0.130:5000"/>
   </FB>
   <FB Name="SEND_SMS" Type="SEND_SMS" Comment="" x="4470.588235294117" y="2470.5882352941176
        ">
      <Parameter Name="ID" Value="10.0.0.130:5000"/>
   </FB>
   <FB Name="E_CYCLE" Type="E_CYCLE" Comment="" x="2470.5882352941176" y="2470.5882352941176"
        >
      <Parameter Name="DT" Value="T#5s"/>
   </FB>
   <FB Name="COMPOSE_ASG" Type="COMPOSE_ASG" Comment="" x="823.5294117647059" y="
        470.5882352941176">
      <Parameter Name="setMag" Value="( f := 12.0)"/>
      <Parameter Name="units" Value="( SIUnit := 5, multiplier := 0 )"/>
   </FB>
   <FB Name="E_DELAY" Type="E_DELAY" Comment="" x="0.0" y="0.0">
      <Parameter Name="DT" Value="T#10s"/>
   </FB>
   <FB Name="DECOMPOSE_DPC" Type="DECOMPOSE_DPC" Comment="" x="4941.176470588235" y="
        1647.0588235294117"/>
   <FB Name="iec_61850_PIOC" Type="iec_61850_PIOC" Comment="" x="2588.235294117647" y="
        117.6470588235294"/>
   <EventConnections>
      <Connection Source="E_DELAY.EO" Destination="COMPOSE_ASG.REQ" Comment="" dx1="0.0" dx2=
           "0.0" dy="0.0"/>
      <Connection Source="iec_61850_PTRC.TRG" Destination="iec_61850_XCBR.REQ" Comment="" dx1
           ="311.7647058823529" dx2="0.0" dy="0.0"/>
      <Connection Source="READ_FROM_DB.CNF" Destination="SEND_SMS.REQ" Comment="" dx1="
           282.35294117647055" dx2="0.0" dy="0.0"/>
      <Connection Source="E_CYCLE.EO" Destination="READ_FROM_DB.REQ" Comment="" dx1="
           223.52941176470586" dx2="0.0" dy="0.0"/>
      <Connection Source="DECOMPOSE_DPC.CNF" Destination="STORE_IN_DB.REQ" Comment="" dx1="
           211.76470588235293" dx2="0.0" dy="0.0"/>
      <Connection Source="iec_61850_XCBR.TRG" Destination="DECOMPOSE_DPC.REQ" Comment="" dx1=
           "223.52941176470586" dx2="70.58823529411764" dy="1035.2941176470588"/>
      <Connection Source="COMPOSE_ASG.CNF" Destination="iec_61850_PIOC.REQ" Comment="" dx1="
           417.6470588235294" dx2="0.0" dy="0.0"/>
      <Connection Source="iec_61850_MMXU.TRG" Destination="iec_61850_PIOC.REQ" Comment="" dx1
           ="411.7647058823529" dx2="0.0" dy="0.0"/>
   </EventConnections>
   <DataConnections>
      <Connection Source="COMPOSE_ASG.asgVal" Destination="iec_61850_PIOC.asgVal" Comment=""
           dx1="541.1764705882352" dx2="0.0" dy="0.0"/>
      <Connection Source="iec_61850_MMXU.MaxAPhs" Destination="iec_61850_PIOC.mvVal" Comment=
           "" dx1="641.1764705882352" dx2="0.0" dy="0.0"/>
      <Connection Source="iec_61850_PTRC.Tr" Destination="iec_61850_XCBR.Tr" Comment="" dx1="
           0.0" dx2="0.0" dy="0.0"/>
      <Connection Source="READ_FROM_DB.Str" Destination="SEND_SMS.Str" Comment="" dx1="
           282.35294117647055" dx2="0.0" dy="0.0"/>
      <Connection Source="DECOMPOSE_DPC.stVal" Destination="STORE_IN_DB.stVal" Comment="" dx1
           ="211.76470588235293" dx2="0.0" dy="0.0"/>
      <Connection Source="DECOMPOSE_DPC.q" Destination="STORE_IN_DB.q" Comment="" dx1="
           211.76470588235293" dx2="0.0" dy="0.0"/>
      <Connection Source="DECOMPOSE_DPC.t" Destination="STORE_IN_DB.t" Comment="" dx1="
           211.76470588235293" dx2="0.0" dy="0.0"/>
      <Connection Source="iec_61850_XCBR.Pos_Out" Destination="DECOMPOSE_DPC.dpcVal" Comment=
           "" dx1="70.58823529411764" dx2="176.47058823529412" dy="447.0588235294117"/>
      <Connection Source="iec_61850_PIOC.Op" Destination="iec_61850_PTRC.Op_In" Comment=""
           dx1="0.0" dx2="0.0" dy="0.0"/>
   </DataConnections>
```

```
      </SubAppNetwork>
   </Application>
   <Device Name="SSLQ7IED2" Type="FORTE_PC" x="0.0" y="0.0">
      <Parameter Name="MGR_ID" Value="&amp;quot;10.0.0.132:61499&amp;quot;"/>
      <Resource Name="SSLQ7IED2_res" Type="EMB_RES" x="0.0" y="0.0">
         <FBNetwork>
            <EventConnections/>
            <DataConnections/>
         </FBNetwork>
      </Resource>
   </Device>
   <Device Name="SSLQ7IED3" Type="FORTE_PC" x="0.0" y="0.0">
      <Parameter Name="MGR_ID" Value="&amp;quot;10.0.0.133:61499&amp;quot;"/>
      <Resource Name="SSLQ7IED3_res" Type="EMB_RES" x="0.0" y="0.0">
         <FBNetwork>
            <FB Name="READ_FROM_DB" Type="READ_FROM_DB" Comment="" x="3411.7647058823527" y="
                2470.5882352941176">
               <Parameter Name="ID" Value="10.0.0.130:5000"/>
            </FB>
            <FB Name="Publish_SEND_SMS" Type="PUBLISH_1" x="0.0" y="0.0">
               <Parameter Name="ID" Value="225.0.0.1:61499"/>
               <Parameter Name="QI" Value="1"/>
            </FB>
            <FB Name="STORE_IN_DB" Type="STORE_IN_DB" Comment="" x="6117.647058823529" y="
                1647.0588235294117">
               <Parameter Name="ID" Value="10.0.0.130:5000"/>
            </FB>
            <FB Name="Subscribe_DECOMPOSE_DPC" Type="SUBSCRIBE_3" x="0.0" y="0.0">
               <Parameter Name="QI" Value="1"/>
               <Parameter Name="ID" Value="225.0.0.2:61499"/>
            </FB>
            <FB Name="E_CYCLE" Type="E_CYCLE" Comment="" x="2470.5882352941176" y="
                2470.5882352941176">
               <Parameter Name="DT" Value="T#5s"/>
            </FB>
            <EventConnections>
               <Connection Source="START.COLD" Destination="READ_FROM_DB.INIT" dx1="0.0" dx2="0.0"
                    dy="0.0"/>
               <Connection Source="START.WARM" Destination="READ_FROM_DB.INIT" dx1="0.0" dx2="0.0"
                    dy="0.0"/>
               <Connection Source="E_CYCLE.EO" Destination="READ_FROM_DB.REQ" Comment="" dx1="
                    223.52941176470586" dx2="0.0" dy="0.0"/>
               <Connection Source="START.COLD" Destination="Publish_SEND_SMS.INIT" dx1="0.0" dx2="
                    0.0" dy="0.0"/>
               <Connection Source="START.WARM" Destination="Publish_SEND_SMS.INIT" dx1="0.0" dx2="
                    0.0" dy="0.0"/>
               <Connection Source="READ_FROM_DB.CNF" Destination="Publish_SEND_SMS.REQ" dx1="0.0"
                    dx2="0.0" dy="0.0"/>
               <Connection Source="START.COLD" Destination="STORE_IN_DB.INIT" dx1="0.0" dx2="0.0"
                    dy="0.0"/>
               <Connection Source="START.WARM" Destination="STORE_IN_DB.INIT" dx1="0.0" dx2="0.0"
                    dy="0.0"/>
               <Connection Source="START.COLD" Destination="Subscribe_DECOMPOSE_DPC.INIT" dx1="0.0"
                     dx2="0.0" dy="0.0"/>
               <Connection Source="START.WARM" Destination="Subscribe_DECOMPOSE_DPC.INIT" dx1="0.0"
                     dx2="0.0" dy="0.0"/>
               <Connection Source="Subscribe_DECOMPOSE_DPC.IND" Destination="STORE_IN_DB.REQ" dx1="
                    0.0" dx2="0.0" dy="0.0"/>
               <Connection Source="START.COLD" Destination="E_CYCLE.START" dx1="0.0" dx2="0.0" dy="
                    0.0"/>
               <Connection Source="START.WARM" Destination="E_CYCLE.START" dx1="0.0" dx2="0.0" dy="
                    0.0"/>
            </EventConnections>
            <DataConnections>
               <Connection Source="READ_FROM_DB.Str" Destination="Publish_SEND_SMS.SD_1" dx1="0.0"
                    dx2="0.0" dy="0.0"/>
               <Connection Source="Subscribe_DECOMPOSE_DPC.RD_1" Destination="STORE_IN_DB.stVal"
                    dx1="0.0" dx2="0.0" dy="0.0"/>
               <Connection Source="Subscribe_DECOMPOSE_DPC.RD_2" Destination="STORE_IN_DB.q" dx1="
                    0.0" dx2="0.0" dy="0.0"/>
               <Connection Source="Subscribe_DECOMPOSE_DPC.RD_3" Destination="STORE_IN_DB.t" dx1="
                    0.0" dx2="0.0" dy="0.0"/>
            </DataConnections>
```

```
      </FBNetwork>
   </Resource>
</Device>
<Device Name="SSLQ7IED4" Type="FORTE_PC" x="0.0" y="0.0">
   <Parameter Name="MGR_ID" Value="&amp;quot;10.0.0.134:61499&amp;quot;"/>
   <Resource Name="SSLQ7IED4_res" Type="EMB_RES" x="0.0" y="0.0">
      <FBNetwork>
         <FB Name="SEND_SMS" Type="SEND_SMS" Comment="" x="4470.588235294117" y="
              2470.5882352941176">
            <Parameter Name="ID" Value="10.0.0.130:5000"/>
         </FB>
         <FB Name="Subscribe_READ_FROM_DB" Type="SUBSCRIBE_1" x="0.0" y="0.0">
            <Parameter Name="ID" Value="225.0.0.1:61499"/>
            <Parameter Name="QI" Value="1"/>
         </FB>
         <FB Name="COMPOSE_ASG" Type="COMPOSE_ASG" Comment="" x="823.5294117647059" y="
              470.5882352941176">
            <Parameter Name="setMag" Value="( f := 12.0)"/>
            <Parameter Name="units" Value="( SIUnit := 5, multiplier := 0 )"/>
         </FB>
         <FB Name="Publish_iec_61850_PIOC" Type="Pub_GOOSE_iec_61850_asg" x="0.0" y="0.0">
            <Parameter Name="ID" Value="SSLQ7IED2LD1/PIOC1.StrVal.setMag.f"/>
            <Parameter Name="Ip" Value="10.0.0.132"/>
         </FB>
         <FB Name="E_DELAY" Type="E_DELAY" Comment="" x="0.0" y="0.0">
            <Parameter Name="DT" Value="T#10s"/>
         </FB>
         <FB Name="DECOMPOSE_DPC" Type="DECOMPOSE_DPC" Comment="" x="4941.176470588235" y="
              1647.0588235294117"/>
         <FB Name="Publish_STORE_IN_DB" Type="PUBLISH_3" x="0.0" y="0.0">
            <Parameter Name="ID" Value="225.0.0.2:61499"/>
            <Parameter Name="QI" Value="1"/>
         </FB>
         <FB Name="Subscribe_iec_61850_XCBR" Type="Sub_GOOSE_iec_61850_dpc" x="0.0" y="0.0">
            <Parameter Name="ID" Value="SSLQ7IED1LD1/LLN0$GO$gcbEventsXcbrPos"/>
            <Parameter Name="Mac" Value="02-42-c0-a8-84-21"/>
         </FB>
         <EventConnections>
            <Connection Source="START.COLD" Destination="SEND_SMS.INIT" dx1="0.0" dx2="0.0" dy="
                 0.0"/>
            <Connection Source="START.WARM" Destination="SEND_SMS.INIT" dx1="0.0" dx2="0.0" dy="
                 0.0"/>
            <Connection Source="START.COLD" Destination="Subscribe_READ_FROM_DB.INIT" dx1="0.0"
                 dx2="0.0" dy="0.0"/>
            <Connection Source="START.WARM" Destination="Subscribe_READ_FROM_DB.INIT" dx1="0.0"
                 dx2="0.0" dy="0.0"/>
            <Connection Source="Subscribe_READ_FROM_DB.IND" Destination="SEND_SMS.REQ" dx1="0.0"
                  dx2="0.0" dy="0.0"/>
            <Connection Source="E_DELAY.EO" Destination="COMPOSE_ASG.REQ" Comment="" dx1="0.0"
                 dx2="0.0" dy="0.0"/>
            <Connection Source="START.COLD" Destination="Publish_iec_61850_PIOC.INIT" dx1="0.0"
                 dx2="0.0" dy="0.0"/>
            <Connection Source="START.WARM" Destination="Publish_iec_61850_PIOC.INIT" dx1="0.0"
                 dx2="0.0" dy="0.0"/>
            <Connection Source="COMPOSE_ASG.CNF" Destination="Publish_iec_61850_PIOC.REQ" dx1="
                 0.0" dx2="0.0" dy="0.0"/>
            <Connection Source="START.COLD" Destination="E_DELAY.START" dx1="0.0" dx2="0.0" dy="
                 0.0"/>
            <Connection Source="START.WARM" Destination="E_DELAY.START" dx1="0.0" dx2="0.0" dy="
                 0.0"/>
            <Connection Source="START.COLD" Destination="Publish_STORE_IN_DB.INIT" dx1="0.0" dx2
                 ="0.0" dy="0.0"/>
            <Connection Source="START.WARM" Destination="Publish_STORE_IN_DB.INIT" dx1="0.0" dx2
                 ="0.0" dy="0.0"/>
            <Connection Source="DECOMPOSE_DPC.CNF" Destination="Publish_STORE_IN_DB.REQ" dx1="
                 0.0" dx2="0.0" dy="0.0"/>
            <Connection Source="START.COLD" Destination="Subscribe_iec_61850_XCBR.INIT" dx1="0.0
                 " dx2="0.0" dy="0.0"/>
            <Connection Source="START.WARM" Destination="Subscribe_iec_61850_XCBR.INIT" dx1="0.0
                 " dx2="0.0" dy="0.0"/>
            <Connection Source="Subscribe_iec_61850_XCBR.IND" Destination="DECOMPOSE_DPC.REQ"
                 dx1="0.0" dx2="0.0" dy="0.0"/>
         </EventConnections>
```

```
            <DataConnections>
               <Connection Source="Subscribe_READ_FROM_DB.RD_1" Destination="SEND_SMS.Str" dx1="0.0
                    " dx2="0.0" dy="0.0"/>
               <Connection Source="COMPOSE_ASG.asgVal" Destination="Publish_iec_61850_PIOC.SD_1"
                    dx1="0.0" dx2="0.0" dy="0.0"/>
               <Connection Source="DECOMPOSE_DPC.stVal" Destination="Publish_STORE_IN_DB.SD_1" dx1=
                    "0.0" dx2="0.0" dy="0.0"/>
               <Connection Source="DECOMPOSE_DPC.q" Destination="Publish_STORE_IN_DB.SD_2" dx1="0.0
                    " dx2="0.0" dy="0.0"/>
               <Connection Source="DECOMPOSE_DPC.t" Destination="Publish_STORE_IN_DB.SD_3" dx1="0.0
                    " dx2="0.0" dy="0.0"/>
               <Connection Source="Subscribe_iec_61850_XCBR.RD_1" Destination="DECOMPOSE_DPC.dpcVal
                    " dx1="0.0" dx2="0.0" dy="0.0"/>
            </DataConnections>
         </FBNetwork>
      </Resource>
   </Device>
   <Device Name="SSLQ7IED1" Type="FORTE_PC" x="0.0" y="0.0">
      <Parameter Name="MGR_ID" Value="&amp;quot;10.0.0.131:61499&amp;quot;"/>
      <Resource Name="SSLQ7IED1_res" Type="EMB_RES" x="0.0" y="0.0">
         <FBNetwork>
            <EventConnections/>
            <DataConnections/>
         </FBNetwork>
      </Resource>
   </Device>
   <Mapping From="diplomarbeitApp.READ_FROM_DB" To="SSLQ7IED3.SSLQ7IED3_res.READ_FROM_DB"/>
   <Mapping From="diplomarbeitApp.STORE_IN_DB" To="SSLQ7IED3.SSLQ7IED3_res.STORE_IN_DB"/>
   <Mapping From="diplomarbeitApp.E_CYCLE" To="SSLQ7IED3.SSLQ7IED3_res.E_CYCLE"/>
   <Mapping From="diplomarbeitApp.SEND_SMS" To="SSLQ7IED4.SSLQ7IED4_res.SEND_SMS"/>
   <Mapping From="diplomarbeitApp.COMPOSE_ASG" To="SSLQ7IED4.SSLQ7IED4_res.COMPOSE_ASG"/>
   <Mapping From="diplomarbeitApp.E_DELAY" To="SSLQ7IED4.SSLQ7IED4_res.E_DELAY"/>
   <Mapping From="diplomarbeitApp.DECOMPOSE_DPC" To="SSLQ7IED4.SSLQ7IED4_res.DECOMPOSE_DPC"/>
   <Segment Name="Ethernet" Type="Ethernet" x="0.0" y="0.0" dx1="0.0"/>
   <Link SegmentName="Ethernet" CommResource="SSLQ7IED2"/>
   <Link SegmentName="Ethernet" CommResource="SSLQ7IED3"/>
   <Link SegmentName="Ethernet" CommResource="SSLQ7IED4"/>
   <Link SegmentName="Ethernet" CommResource="SSLQ7IED1"/>
</System>
```

Listing 7.75: IEC 61499 System Output File

7.6.4 Forte Boot Files

```
;<Request ID="0" Action="CREATE"><FB Name="SSLQ7IED1_res" Type="EMB_RES" /></Request>
SSLQ7IED1_res;<Request ID="1" Action="START"/>
```

Listing 7.76: Forte Boot Files for IED 1

```
;<Request ID="0" Action="CREATE"><FB Name="SSLQ7IED2_res" Type="EMB_RES" /></Request>
SSLQ7IED2_res;<Request ID="1" Action="START"/>
```

Listing 7.77: Forte Boot Files for IED 2

```
;<Request ID="0" Action="CREATE"><FB Name="SSLQ7IED3_res" Type="EMB_RES" /></Request>
SSLQ7IED3_res;<Request ID="1" Action="CREATE"><FB Name="READ_FROM_DB" Type="READ_FROM_DB" /></
     Request>
SSLQ7IED3_res;<Request ID="2" Action="WRITE"><Connection Source="10.0.0.130:5000" Destination="
     READ_FROM_DB.ID" /></Request>
SSLQ7IED3_res;<Request ID="3" Action="CREATE"><FB Name="Publish_SEND_SMS" Type="PUBLISH_1" /></
     Request>
SSLQ7IED3_res;<Request ID="4" Action="WRITE"><Connection Source="225.0.0.1:61499" Destination="
     Publish_SEND_SMS.ID" /></Request>
SSLQ7IED3_res;<Request ID="5" Action="WRITE"><Connection Source="1" Destination="Publish_SEND_SMS.
     QI" /></Request>
SSLQ7IED3_res;<Request ID="6" Action="CREATE"><FB Name="STORE_IN_DB" Type="STORE_IN_DB" /></Request
     >
SSLQ7IED3_res;<Request ID="7" Action="WRITE"><Connection Source="10.0.0.130:5000" Destination="
     STORE_IN_DB.ID" /></Request>
```

```
SSLQ7IED3_res;<Request ID="8" Action="CREATE"><FB Name="Subscribe_DECOMPOSE_DPC" Type="SUBSCRIBE_3"
     /></Request>
SSLQ7IED3_res;<Request ID="9" Action="WRITE"><Connection Source="1" Destination="
    Subscribe_DECOMPOSE_DPC.QI" /></Request>
SSLQ7IED3_res;<Request ID="10" Action="WRITE"><Connection Source="225.0.0.2:61499" Destination="
    Subscribe_DECOMPOSE_DPC.ID" /></Request>
SSLQ7IED3_res;<Request ID="11" Action="CREATE"><FB Name="E_CYCLE" Type="E_CYCLE" /></Request>
SSLQ7IED3_res;<Request ID="12" Action="WRITE"><Connection Source="T#5s" Destination="E_CYCLE.DT" />
    </Request>
SSLQ7IED3_res;<Request ID="13" Action="CREATE"><Connection Source="START.COLD" Destination="
    READ_FROM_DB.INIT" /></Request>
SSLQ7IED3_res;<Request ID="14" Action="CREATE"><Connection Source="START.WARM" Destination="
    READ_FROM_DB.INIT" /></Request>
SSLQ7IED3_res;<Request ID="15" Action="CREATE"><Connection Source="E_CYCLE.EO" Destination="
    READ_FROM_DB.REQ" /></Request>
SSLQ7IED3_res;<Request ID="16" Action="CREATE"><Connection Source="START.COLD" Destination="
    Publish_SEND_SMS.INIT" /></Request>
SSLQ7IED3_res;<Request ID="17" Action="CREATE"><Connection Source="START.WARM" Destination="
    Publish_SEND_SMS.INIT" /></Request>
SSLQ7IED3_res;<Request ID="18" Action="CREATE"><Connection Source="READ_FROM_DB.CNF" Destination="
    Publish_SEND_SMS.REQ" /></Request>
SSLQ7IED3_res;<Request ID="19" Action="CREATE"><Connection Source="START.COLD" Destination="
    STORE_IN_DB.INIT" /></Request>
SSLQ7IED3_res;<Request ID="20" Action="CREATE"><Connection Source="START.WARM" Destination="
    STORE_IN_DB.INIT" /></Request>
SSLQ7IED3_res;<Request ID="21" Action="CREATE"><Connection Source="START.COLD" Destination="
    Subscribe_DECOMPOSE_DPC.INIT" /></Request>
SSLQ7IED3_res;<Request ID="22" Action="CREATE"><Connection Source="START.WARM" Destination="
    Subscribe_DECOMPOSE_DPC.INIT" /></Request>
SSLQ7IED3_res;<Request ID="23" Action="CREATE"><Connection Source="Subscribe_DECOMPOSE_DPC.IND"
    Destination="STORE_IN_DB.REQ" /></Request>
SSLQ7IED3_res;<Request ID="24" Action="CREATE"><Connection Source="START.COLD" Destination="E_CYCLE.
    START" /></Request>
SSLQ7IED3_res;<Request ID="25" Action="CREATE"><Connection Source="START.WARM" Destination="E_CYCLE.
    START" /></Request>
SSLQ7IED3_res;<Request ID="26" Action="CREATE"><Connection Source="READ_FROM_DB.Str" Destination="
    Publish_SEND_SMS.SD_1" /></Request>
SSLQ7IED3_res;<Request ID="27" Action="CREATE"><Connection Source="Subscribe_DECOMPOSE_DPC.RD_1"
    Destination="STORE_IN_DB.stVal" /></Request>
SSLQ7IED3_res;<Request ID="28" Action="CREATE"><Connection Source="Subscribe_DECOMPOSE_DPC.RD_2"
    Destination="STORE_IN_DB.q" /></Request>
SSLQ7IED3_res;<Request ID="29" Action="CREATE"><Connection Source="Subscribe_DECOMPOSE_DPC.RD_3"
    Destination="STORE_IN_DB.t" /></Request>
SSLQ7IED3_res;<Request ID="30" Action="START"/>
```

Listing 7.78: Forte Boot Files for IED 3

```
;<Request ID="0" Action="CREATE"><FB Name="SSLQ7IED4_res" Type="EMB_RES" /></Request>
SSLQ7IED4_res;<Request ID="1" Action="CREATE"><FB Name="SEND_SMS" Type="SEND_SMS" /></Request>
SSLQ7IED4_res;<Request ID="2" Action="WRITE"><Connection Source="10.0.0.130:5000" Destination="
    SEND_SMS.ID" /></Request>
SSLQ7IED4_res;<Request ID="3" Action="CREATE"><FB Name="Subscribe_READ_FROM_DB" Type="SUBSCRIBE_1"
    /></Request>
SSLQ7IED4_res;<Request ID="4" Action="WRITE"><Connection Source="225.0.0.1:61499" Destination="
    Subscribe_READ_FROM_DB.ID" /></Request>
SSLQ7IED4_res;<Request ID="5" Action="WRITE"><Connection Source="1" Destination="
    Subscribe_READ_FROM_DB.QI" /></Request>
SSLQ7IED4_res;<Request ID="6" Action="CREATE"><FB Name="COMPOSE_ASG" Type="COMPOSE_ASG" /></Request
    >
SSLQ7IED4_res;<Request ID="7" Action="WRITE"><Connection Source="( f := 12.0)" Destination="
    COMPOSE_ASG.setMag" /></Request>
SSLQ7IED4_res;<Request ID="8" Action="WRITE"><Connection Source="( SIUnit := 5, multiplier := 0 )"
    Destination="COMPOSE_ASG.units" /></Request>
SSLQ7IED4_res;<Request ID="9" Action="CREATE"><FB Name="Publish_iec_61850_PIOC" Type="
    Pub_GOOSE_iec_61850_asg" /></Request>
SSLQ7IED4_res;<Request ID="10" Action="WRITE"><Connection Source="SSLQ7IED2LD1/PIOC1.StrVal.setMag.
    f" Destination="Publish_iec_61850_PIOC.ID" /></Request>
SSLQ7IED4_res;<Request ID="11" Action="WRITE"><Connection Source="10.0.0.132" Destination="
    Publish_iec_61850_PIOC.Ip" /></Request>
SSLQ7IED4_res;<Request ID="12" Action="CREATE"><FB Name="E_DELAY" Type="E_DELAY" /></Request>
SSLQ7IED4_res;<Request ID="13" Action="WRITE"><Connection Source="T#10s" Destination="E_DELAY.DT" /
    ></Request>
```

```
SSLQ7IED4_res;<Request ID="14" Action="CREATE"><FB Name="DECOMPOSE_DPC" Type="DECOMPOSE_DPC" /></
    Request>
SSLQ7IED4_res;<Request ID="15" Action="CREATE"><FB Name="Publish_STORE_IN_DB" Type="PUBLISH_3" /></
    Request>
SSLQ7IED4_res;<Request ID="16" Action="WRITE"><Connection Source="225.0.0.2:61499" Destination="
    Publish_STORE_IN_DB.ID" /></Request>
SSLQ7IED4_res;<Request ID="17" Action="WRITE"><Connection Source="1" Destination="
    Publish_STORE_IN_DB.QI" /></Request>
SSLQ7IED4_res;<Request ID="18" Action="CREATE"><FB Name="Subscribe_iec_61850_XCBR" Type="
    Sub_GOOSE_iec_61850_dpc" /></Request>
SSLQ7IED4_res;<Request ID="19" Action="WRITE"><Connection Source="SSLQ7IED1LD1/
    LLN0$GO$gcbEventsXcbrPos" Destination="Subscribe_iec_61850_XCBR.ID" /></Request>
SSLQ7IED4_res;<Request ID="20" Action="WRITE"><Connection Source="02-42-c0-a8-84-21" Destination="
    Subscribe_iec_61850_XCBR.Mac" /></Request>
SSLQ7IED4_res;<Request ID="21" Action="CREATE"><Connection Source="START.COLD" Destination="
    SEND_SMS.INIT" /></Request>
SSLQ7IED4_res;<Request ID="22" Action="CREATE"><Connection Source="START.WARM" Destination="
    SEND_SMS.INIT" /></Request>
SSLQ7IED4_res;<Request ID="23" Action="CREATE"><Connection Source="START.COLD" Destination="
    Subscribe_READ_FROM_DB.INIT" /></Request>
SSLQ7IED4_res;<Request ID="24" Action="CREATE"><Connection Source="START.WARM" Destination="
    Subscribe_READ_FROM_DB.INIT" /></Request>
SSLQ7IED4_res;<Request ID="25" Action="CREATE"><Connection Source="Subscribe_READ_FROM_DB.IND"
    Destination="SEND_SMS.REQ" /></Request>
SSLQ7IED4_res;<Request ID="26" Action="CREATE"><Connection Source="E_DELAY.EO" Destination="
    COMPOSE_ASG.REQ" /></Request>
SSLQ7IED4_res;<Request ID="27" Action="CREATE"><Connection Source="START.COLD" Destination="
    Publish_iec_61850_PIOC.INIT" /></Request>
SSLQ7IED4_res;<Request ID="28" Action="CREATE"><Connection Source="START.WARM" Destination="
    Publish_iec_61850_PIOC.INIT" /></Request>
SSLQ7IED4_res;<Request ID="29" Action="CREATE"><Connection Source="COMPOSE_ASG.CNF" Destination="
    Publish_iec_61850_PIOC.REQ" /></Request>
SSLQ7IED4_res;<Request ID="30" Action="CREATE"><Connection Source="START.COLD" Destination="E_DELAY.
    START" /></Request>
SSLQ7IED4_res;<Request ID="31" Action="CREATE"><Connection Source="START.WARM" Destination="E_DELAY.
    START" /></Request>
SSLQ7IED4_res;<Request ID="32" Action="CREATE"><Connection Source="START.COLD" Destination="
    Publish_STORE_IN_DB.INIT" /></Request>
SSLQ7IED4_res;<Request ID="33" Action="CREATE"><Connection Source="START.WARM" Destination="
    Publish_STORE_IN_DB.INIT" /></Request>
SSLQ7IED4_res;<Request ID="34" Action="CREATE"><Connection Source="DECOMPOSE_DPC.CNF" Destination="
    Publish_STORE_IN_DB.REQ" /></Request>
SSLQ7IED4_res;<Request ID="35" Action="CREATE"><Connection Source="START.COLD" Destination="
    Subscribe_iec_61850_XCBR.INIT" /></Request>
SSLQ7IED4_res;<Request ID="36" Action="CREATE"><Connection Source="START.WARM" Destination="
    Subscribe_iec_61850_XCBR.INIT" /></Request>
SSLQ7IED4_res;<Request ID="37" Action="CREATE"><Connection Source="Subscribe_iec_61850_XCBR.IND"
    Destination="DECOMPOSE_DPC.REQ" /></Request>
SSLQ7IED4_res;<Request ID="38" Action="CREATE"><Connection Source="Subscribe_READ_FROM_DB.RD_1"
    Destination="SEND_SMS.Str" /></Request>
SSLQ7IED4_res;<Request ID="39" Action="CREATE"><Connection Source="COMPOSE_ASG.asgVal" Destination=
    "Publish_iec_61850_PIOC.SD_1" /></Request>
SSLQ7IED4_res;<Request ID="40" Action="CREATE"><Connection Source="DECOMPOSE_DPC.stVal" Destination
    ="Publish_STORE_IN_DB.SD_1" /></Request>
SSLQ7IED4_res;<Request ID="41" Action="CREATE"><Connection Source="DECOMPOSE_DPC.q" Destination="
    Publish_STORE_IN_DB.SD_2" /></Request>
SSLQ7IED4_res;<Request ID="42" Action="CREATE"><Connection Source="DECOMPOSE_DPC.t" Destination="
    Publish_STORE_IN_DB.SD_3" /></Request>
SSLQ7IED4_res;<Request ID="43" Action="CREATE"><Connection Source="Subscribe_iec_61850_XCBR.RD_1"
    Destination="DECOMPOSE_DPC.dpcVal" /></Request>
SSLQ7IED4_res;<Request ID="44" Action="START"/>
```

Listing 7.79: Forte Boot Files for IED 4

7.7 Testbed Files

7.7.1 Docker Compose File

```
version: '2.1'
```

```
services:
  ied1:
    build:
      context: .
      dockerfile: ied1.Dockerfile
    networks:
      macvlan:
        ipv4_address: 10.0.0.131
    restart: always
    cap_add:
      - NET_ADMIN
    mac_address: 02:42:c0:a8:84:21
    ports:
      - "102"
      - "61499"

  ied2:
    build:
      context: .
      dockerfile: ied2.Dockerfile
    networks:
      macvlan:
        ipv4_address: 10.0.0.132
    restart: always
    cap_add:
      - NET_ADMIN
    mac_address: 02:42:c0:a8:84:22
    ports:
      - "102"
      - "61499"

  ied3:
    build:
      context: .
      dockerfile: ied3.Dockerfile
    networks:
      macvlan:
        ipv4_address: 10.0.0.133
    restart: always
    cap_add:
      - NET_ADMIN
    mac_address: 02:42:c0:a8:84:23
    ports:
      - "102"
      - "61499"

  ied4:
    build:
      context: .
      dockerfile: ied4.Dockerfile
    networks:
      macvlan:
        ipv4_address: 10.0.0.134
    restart: always
    cap_add:
      - NET_ADMIN
    mac_address: 02:42:c0:a8:84:24
    ports:
    - "102"
    - "61499"

networks:
  macvlan:
    driver: macvlan
    ipam:
      driver: default
      config:
        - subnet: 10.0.0.0/24
          gateway: 10.0.0.138
    driver_opts:
      parent: enp0s31f6
```

Listing 7.80: Docker-Compose File

List of Figures

List of Tables

List of Algorithms

Acronyms

ASCI Abstract Communication Service Interface. 21, 22

ASG Analogue Setting. 43

AUDRAGA Automatische Installation drahtloser Systeme der Gebäudeautomation. 27

AUTEG Automatisierter Entwurf für die Gebäudeautomation. 27, 33

AUTERAS Automatisierter Entwurf von Raumautomationssystemen. 27, 33, 78

BA Building Automation. 27, 33

BFB Basic Function Block. 16, 42, 44

CDC Common Data Class. 21, 26, 42, 43, 45, 67, 83, 84, 87

CEN Comité Européen de Normalisation. 8

CENELEC Comité Européen de Normalisation Electrotechnique. 8

CFB Composite Function Block. 16, 17, 26, 44

CID Configured IED Description. 24, 37, 65, 73

CIM Common Information Model. 11, 12, 18, 25

COSEM Companion Specification for Energy Metering. 25

DA Data Attribute. 46, 51, 53, 58, 78

DER Distributed Energy Resource. 1

DMS Distribution Management Systems. 10, 11

DO Data Object. 42, 43, 46, 51, 53, 58, 78

DPC Double Point Control. 21, 43

DTD Document Type Definition. 47–49, 56

ECC Electronic Control Chart. 15, 16

EMS Energy Management System. 10, 11

EPRI Electric Power Research Institute. 10, 11

ETSI European Telecommunications Standard Institute. 8

FB Function Block. 3, 4, 14–17, 26, 27, 30, 31, 33–39, 41–49, 53–56, 58–60, 63, 65, 67–69, 71–73, 77–79, 87

FBD Function Block Diagram. 13, 15

FC Functional Constraint. 42, 45, 87

FFG Austrian Research Promotion Agency. 2

FLISR Fault Location Isolation and Service Restoration. 1, 3, 29, 34, 67, 78

GMD Grid Measurement Device. 67

GOOSE Generic Object Oriented Substation Event. 22–24, 37, 63, 73, 78

GSSE Generic Substation Status Events. 22, 23, 78

HVAC Heating, Ventilation and Air-Conditioning. 27

ICD IED Capabilities Description. 23, 50

ICT Information and Communication. 1, 4, 28

IDE Integrated Development Environment. 63

IEC International Electrotechnical Commission. 3–5, 8, 10–15, 18, 19, 22, 25–33, 36–39, 41–43, 45–51, 53, 54, 56, 58, 63, 65, 67–69, 71, 73, 77–79, 83, 87

IED Intelligent Electronic Device. 19–21, 23, 24, 27, 46, 50, 51, 53, 54, 56, 58, 60, 62, 65, 70–74

IID Instantiated IED Description. 24

IL Instruction List. 13

iLN intelligent Logical Node. 26

IP Internet Protocol. 63, 68, 74

JAXB Java Architecture for XML Binding. 56

LD Ladder Diagram. 13, 20–22, 51

LN Logical Node. 20–24, 26, 27, 31, 34, 39, 42, 45, 46, 51, 53, 54, 58, 70, 78, 87

MAC Media Access Control. 63, 74

MMS Manufacturing Messaging Specification. 22, 23

MU Merging Unit. 23

PD Physical Device. 19–21

PLC Programmable Logic Controller. 12–15

PoSyCo Power System Cognification. 2, 10

PSAL Power System Automation Language. 28, 77

PV Photovoltaic. 1

RDF Resource Description Framework. 11

RES Renewable Energy Source. 1, 2

SCADA Supervisory Control and Data Acquisition. 23, 25

SCD Substation Configuration Description. 24, 25, 27, 38, 54

SCL System Configuration Language. 23, 24, 37, 50, 56, 62, 65, 69, 70, 73

SCUBA Self-organising, Cooperative, and robUst Building Automation. 27

SED System Exchange Description. 24

SFC Sequential Function Chart. 13

SG Smart Grid. 1–5, 7, 8, 10, 12, 15, 26, 28–33, 38–41, 47, 49, 54–56, 67, 77, 78

SGAM Smart Grid Architecture Model. 8, 10, 25, 77

SIFB Service Interface Function Block. 17, 42, 44

SMS Short Message Service. 68, 73, 74

SMV Sampled Measured Value. 22, 23, 78

SNTP Simple Network Time Protocol. 22

SPS Single Point Status. 21

SSD System Specification Description. 23, 50, 69

ST Structured Text. 13

TC Technical Committee. 18, 25

TOPAS Tools for Continuous Building Performance Auditing. 27

TR Technical Report. 19, 25

UML Unified Modelling Language. 11, 25, 56

UUID Universally Unique Identifier. 60

VDI Verein Deutscher Ingenieure. 27

XML eXtensible Markup Language. 11, 23, 41, 43, 44, 47–51, 56, 60, 62, 65, 68, 69, 73

XSD XML Schema Definition. 47, 50, 56, 60, 65

Bibliography

[ABS] Filip Andren, Roland Brundlinger, and Thomas Strasser. Iec 61850/61499 control of distributed energy resources: Concept, guidelines, and implementation.

[ASK14] Filip Andrén, Thomas Strasser, and Wolfgang Kastner. Model-driven engineering applied to smart grid automation using iec 61850 and iec 61499. In *2014 Power Systems Computation Conference*, pages 1–7. IEEE, 2014.

[ASS] Filip Andrén, Matthias Stifter, and Thomas Strasser. Towards a semantic driven framework for smart grid applications: Model-driven development using cim, iec 61850 and iec 61499; organ der gesellschaft für informatik e.v. und mit ihr assoziierter organisationen.

[Atl18] Energy Atlas. Energy atlas 2018—facts and figures about renewables in europe; heinrich böll foundation. *Friends of the Earth Europe*, 2018.

[BBBW03] Rudolf P. Baumann, Klaus-Peter Brand, Christoph Brunner, and Wolfgang Wimmer. Der standard iec 61850: Iec 61850 in schaltanlagen als kern einer durchgängigen kommunikationslösung für den netzbetreiber. Number 03, pages 31–37, 2003.

[Ber19] Tom Berry. Introduction to iec 62361-102 cim - 61850 harmonization. 06 2019.

[BLS10] P. Brédillet, Eric Lambert, and Eric Schultz. Cim, 61850, cosem standards used in a model driven integration approach to build the smart grid service oriented architecture. pages 467 – 471, 11 2010.

[Bor16] Stuart Borlase. *Smart grids: infrastructure, technology, and solutions*. CRC press, 2016.

[BS20] Bernd Buchholz and Zbigniew Styczynski. *Smart Grids: Fundamentals and Technologies in Electric Power Systems of the future*. 01 2020.

[BW02] Klaus-Peter Brand and Wolfgang Wimmer. Der standard iec 61850 - offene kommunikation in schaltanlagen im deregulierten strommarkt. volume 93, pages 9–13, 2002.

[C+08] International Electrotechnical Commission et al. Iec 61970: Energy management system application program interface. *International Electrotechnical Commission: Geneva, Switzerland*, 2008.

[C+10] International Electrotechnical Commission et al. Iec 61499: Distributed control reference mode. *International Electrotechnical Commission: Geneva, Switzerland*, 2010.

[C+12a] International Electrotechnical Commission et al. Iec 61131: Programmable controllers. *International Electrotechnical Commission: Geneva, Switzerland*, 2012.

[C+12b] International Electrotechnical Commission et al. Iec 61850: Communication networks and systems for power utility automation. *International Electrotechnical Commission: Geneva, Switzerland*, 2012.

[C+15] International Electrotechnical Commission et al. Iec 62559: Use case methodology. *International Electrotechnical Commission: Geneva, Switzerland*, 2015.

[C+16] International Electrotechnical Commission et al. Iec 62357: Power systems management and associated information exchange. *International Electrotechnical Commission: Geneva, Switzerland*, 2016.

[C+18] International Electrotechnical Commission et al. Iec tr 62361-102:2018 power systems management and associated information exchange - interoperability in the long term - part 102: Cim - iec 61850 harmonization. *International Electrotechnical Commission: Geneva, Switzerland*, 2018.

[CCE12] Smart Grid Coordination CEN-CENELEC-ETSI. Group.(2012). *Smart Grid Reference Architecture*, pages 1–107, 2012.

[CKK19] Stephan Cejka, Felix Knorr, and Florian Kintzler. Privacy issues in smart buildings by examples in smart metering. 06 2019.

[CKKL17] Stephan Cejka, Felix Knorr, Lukas Krammer, and Daniel Lechner. Privacy enhancing technologies for the smart building domain. 03 2017.

[Dil20] G. Dileep. A survey on smart grid technologies and applications. *Renewable Energy*, 146:2589 – 2625, 2020.

[dL19] Francisco Chagas de Lima. Power system protection modelling with iec 61850 and iec 61499. 2019.

[DPK09] Henrik Dibowski, Joern Ploennigs, and Klaus Kabitzsch. Automated design of building automation systems. *IEEE Transactions on Industrial Electronics*, 57(11):3606–3613, 2009.

[DPK10] H. Dibowski, J. Ploennigs, and K. Kabitzsch. Automated design of building automation systems. *IEEE Transactions on Industrial Electronics*, 57(11):3606–3613, 2010.

[FL86] Donald Friesen and Michael Langston. Variable sized bin packing. *SIAM Journal on Computing*, 15:222–230, 02 1986.

[GT] S Grijalva and M U Tariq. Prosumer-based smart grid architecture enables a flat, sustainable electricity industry.

[GUD17] Marion Gottschalk, Mathias Uslar, and Christina Delfs. *The Use Case and Smart Grid Architecture Model Approach - The IEC 62559-2 Use Case Template and the SGAM applied in various domains*. 01 2017.

[HSM+12] I. Hegny, T. Strasser, M. Melik-Merkumians, M. Wenger, and A. Zoitl. Towards an increased reusability of distributed control applications modeled in iec 61499. In *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, pages 1–8, 2012.

[Hua18] Wei Huang. Learn iec 61850 configuration in 30 minutes. In *2018 71st Annual Conference for Protective Relay Engineers (CPRE)*, pages 1–5. IEEE, 2018.

[HVNS11] N. Higgins, V. Vyatkin, N. C. Nair, and K. Schwarz. Distributed power system automation with iec 61850, iec 61499, and intelligent control. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(1):81–92, 2011.

[KYL+11] Yong-Hak Kim, Jeon Yeol, You Lee, Yong An, and In Song. Development of iec61850 based substation engineering tools with iec61850 schema library. *Smart Grid and Renewable Energy*, 02, 01 2011.

[LLP+17] Suzanne Lesecq, Maxime Louvel, Joern Ploennigs, Susan Rea, Nicolas Rehault, Dominik Wystrcil, and Alan McGibney. Topas, an iot driven framework for energy efficiency in buildings. *Proceedings*, 1(7), 2017.

[Mac07] Ralph Mackiewicz. Benefits of iec 61850. *Praxis Profiline–IEC*, 61850, 2007.

[Mac11] R. Mackiewicz. Harmonization of the cim and iec 61850. In *2011 IEEE/PES Power Systems Conference and Exposition*, pages 1–1, 2011.

[McM07] Alan W McMorran. An introduction to iec 61970-301 & 61968-11: The common information model. 2007.

[Mom12] J.A. Momoh. *Smart Grid: Fundamentals of Design and Analysis*. I E E Power Engineering Series. Wiley, 2012.

[MRL+13] A. McGibney, S. Rea, M. Lehmann, S. Thior, S. Lesecq, M. Hendriks, C. Gardeux, L. T. Mai, F. Pacull, J. Ploennigs, T. Basten, and D. Pesch. A systematic engineering tool chain approach for self-organizing building automation systems. In *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, pages 7696–7701, 2013.

[MSS+12] A. W. McMorran, E. M. Stewart, C. M. Shand, S. E. Rudd, and G. A. Taylor. Addressing the challenge of data interoperability for off-line analysis of distribution networks in the smart grid. In *PES T D 2012*, pages 1–5, 2012.

[Neu17] Christian Neureiter. *A Domain-Specific, Model Driven Engineering Approach for Systems Engineering in the Smart Grid*. 11 2017.

[oT15] Austrian Institute of Technology. Projektantrag: Power System Cognification (PoSyCo). Technical report, Austrian Research and Promotion Agency (FFG), 2015.

[PA] Filip Pröstl Andrén. *Model-driven engineering for smart grid automation*. Wien.

[PDRK13] Joern Ploennigs, Herik Dibowski, Uwe Ryssel, and Klaus Kabitzsch. Ganzheitlicher, automatischer entwurf drahtloser gebäudeautomationssysteme. 61(6):393–402, 2013.

[PSK+09] Y. Pradeep, P. Seshuraju, S. A. Khaparde, V. S. Warrier, and S. Cherian. Cim and iec 61850 integration issues: Application to power systems. In *2009 IEEE Power Energy Society General Meeting*, pages 1–6, 2009.

[PVM13] P. Parikh, I. Voloh, and M. Mahony. Fault location, isolation, and service restoration (flisr) technique using iec 61850 goose. In *2013 IEEE Power Energy Society General Meeting*, pages 1–6, 2013.

[SFA18] Kaoutar Saadi, Berna Farraa, and Rosa Abbou. Methodology of conception, implementation and validation of an iec 61850 communication system in smart grids demonstration platform. pages 159–164, 10 2018.

[(SG10] SMB Smart Grid Strategic Group (SG3). Iec smart grid standardization roadmap, 2010.

[SKG16] Daniel Schachinger, Wolfgang Kastner, and Stefan Gaida. Ontology-based abstraction layer for smart grid interaction in building energy management systems. In *2016 IEEE International Energy Conference (ENERGYCON)*, pages 1–6. IEEE, 2016.

[SRS10] R. Santodomingo, J. A. Rodríguez-Mondéjar, and M. A. Sanz-Bobi. Ontology matching approach to the harmonization of cim and iec 61850 standards. In

2010 First IEEE International Conference on Smart Grid Communications, pages 55–60, 2010.

[tec11] Technology roadmap smart grid, 2011.

[VPT+14] V. Vlad, Cezar Popa, C. Turcu, Corneliu, and BUZDUGA. Control architecture for substation automation systems based on iec 61850 and iec 61499 standards. 2014.

[Wim14] Bernhard Wimmer. Iec 61850 edition 2 and engineering. *Pac World*, Dec 2014.

[WMS17] S. Wilker, M. Meisel, and T. Sauter. Smart grid architecture model standardization and the applicability of domain language specific modeling tools. In *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pages 152–157, June 2017.

[YDV17] C. Yang, V. Dubinin, and V. Vyatkin. Ontology driven approach to generate distributed automation control from substation automation design. *IEEE Transactions on Industrial Informatics*, 13(2):668–679, 2017.

[YZV14] Chen-Wei Yang, Gulnara Zhabelova, and Valeriy Vyatkin. Sysgrid: Iec 61850 and iec 61499 standard based engineering tool for smart grid automation design. *EAI Endorsed Transactions on Energy Web*, 1:e5, 12 2014.

[ZAS18] C. Zanabria, F. P. Andrén, and T. I. Strasser. Comparing specification and design approaches for power systems applications. In *2018 IEEE PES Transmission Distribution Conference and Exhibition - Latin America (T D-LA)*, pages 1–5, 2018.

[ZL14] Alois Zoitl and Robert Lewis. *Modelling control systems using IEC 61499. 2nd Edition*. 01 2014.

[ZSD11] Lin Zhu, Dongyuan Shi, and Xianzhong Duan. Standard function blocks for flexible ied in iec 61850-based substation automation. *Power Delivery, IEEE Transactions on*, 26:1101 – 1110, 05 2011.

[ZV12] G. Zhabelova and V. Vyatkin. Multiagent smart grid automation architecture based on iec 61850/61499 intelligent logical nodes. *IEEE Transactions on Industrial Electronics*, 59(5):2351–2362, 2012.

[ZYV13] Gulnara Zhabelova, Chen-Wei Yang, and Valeriy Vyatkin. Sysgrid: Iec 61850/iec 61499 based engineering process for smart grid automation design. pages 364–369, 07 2013.