LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
LUT School of Energy Systems
Electrical Engineering

Petra Raussi

REAL-TIME LABORATORY INTERCONNECTION FOR SMART GRID TESTING

Examiners: Prof. Samuli Honkapuro
D.Sc. Anna Kulmala

ABSTRACT

Lappeenranta University of Technology
LUT School of Energy Systems
Electrical Engineering

Petra Raussi
Real-time laboratory interconnection for smart grid testing
Master's thesis
2017
101 pages, 38 pictures, 8 tables and 4 appendixes

Examiners: Professor Samuli Honkapuro
D.Sc. Anna Kulmala

Keywords: IEC 61850, Common Information Model, smart grids, real-time laboratory interconnection, delay testing

Future renewable energy systems, which are based on decentralized energy production, are smart grids that embody advanced automation and communication technologies to maintain the level of reliability and power quality present in traditional centralized power systems. In this thesis, a real-time laboratory interconnection proposed in the European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out (ERIGrid) project was implemented in Multipower laboratory at VTT Technical Research Centre of Finland. The main goal of ERIGrid is to develop integrated research infrastructure for smart grids.

A literature review of relevant remote connections and interconnections conducted previously are presented. Relevant standards IEC 61850 and Common Information Model (CIM) and their harmonization are discussed. The case laboratory and Joint Research Facility for Smart Energy Networks with Distributed Energy Resources (JaNDER), a concept for implementing interconnection between laboratories developed in the ERIGrid project, are portrayed. Based on the information presented the real-time laboratory interconnection was implemented and testing was conducted to obtain possible example delays for data transfer via the interconnection.

The key result of this thesis is that the implementation of the real-time laboratory interconnection was successful. Possible example delays were obtained during the testing, but further testing is needed to obtain statistical delay values.

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
LUT School of Energy Systems
Sähkötekniikka

Petra Raussi
Laboratorioiden reaaliaikainen yhteiskäyttö älyverkkotestauksessa
Diplomityö
2017
101 sivua, 38 kuvaa, 8 taulukkoa ja 4 liitettä

Tarkastajat: Professori Samuli Honkapuro
TkT Anna Kulmala

Hakusanat: IEC 61850, Common Information Model, älyverkot, laboratorioiden reaaliaikainen yhteiskäyttö, viivetestaus

Tulevaisuuden hajautettuun energiantuotantoon perustuvat uusiutuvat energiajärjestelmät ovat älyverkkoja, jotka käsittävät kehittyneitä automaatio- ja tiedonsiirtoteknologioita sähkön laadun ja luotettavuuden ylläpitämiseksi perinteisiä keskitettyjä sähköjärjestelmiä vastaavalla tasolla. Tässä diplomityössä toteutettiin ERIGrid-projektissa ehdotettu laboratorioiden reaaliaikainen yhteiskäyttö Teknologian Tutkimuskeskus VTT:n Multipower-laboratoriossa. ERIGrid-projektin (European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out) päätavoitteena on kehittää integroitu tutkimusinfrastruktuuri älyverkoille.

Kirjallisuustutkimuksessa käsitellään aiemmin toteutetut olennaiset etäyhteydet ja yhteiskäytöt. Työssä käsitellään olennaiset standardit IEC 61850 ja Common Information Model (CIM) ja niiden yhtenäistäminen. Työssä kuvataan myös case laboratorio ja JaNDER (Joint Research Facility for Smart Energy Networks with Distributed Energy Resources), joka on ERIGrid-projektissa kehitetty laboratorioiden yhteiskäytön toteutuskonsepti. Laboratorioiden reaaliaikainen yhteiskäyttö toteutettiin perustuen työssä käsiteltyyn informaatioon, lisäksi toteutettiin testaus yhteiskäytön tiedonsiirron mahdollisten esimerkkiviiveiden saamiseksi.

Tämän diplomityön keskeinen tulos on se, että laboratorioiden reaaliaikaisen yhteiskäytön toteutus oli onnistunut. Mahdollisia esimerkkiviiveitä saatiin kirjattua testauksen aikana, mutta tilastollisien viivearvojen hankkimiseksi tarvitaan lisätestejä.

ACKNOWLEDGEMENTS

This thesis work was completed in Power Systems and Renewables Team at VTT Technical Research Centre of Finland Ltd. The research study was part of the ERIGrid project funded by the European Union.

First, I would like to thank my supervisors Prof. Samuli Honkapuro and D.Sc. Anna Kulmala for their guidance throughout my time writing the thesis and for being always available to provide feedback and suggestions. I feel very fortunate to have had such encouraging and supportive supervisors. I am very thankful for D.Sc. Kari Mäki presenting me the opportunity to do the thesis on the ERIGrid project at VTT. I would also like to thank Riku Pasonen and Jarmo Kuusisto for sharing their vast knowledge on Multipower laboratory and helping me with the testing, and Jarno Karu for helping correct the routings at the laboratory. Additionally, many thanks to Geert-Jan Bluemink and all my colleagues at VTT for creating an inspiring working environment and challenging me to see the thesis from several perspectives through conversations.

I would like to thank Daniele Pala of RSE SpA for his guidance in connecting local Redis, cloud Redis, and ABB COM600 and for helping me with the testing. I would also like to thank Markku Hamarila of ABB for answering all my doubts regarding ABB COM600. Many thanks to D.Sc. Konstantin Ignatiev for his support in the search for references and PhD Hanna Niemelä for helping in formatting the references and the entire thesis. However, only I am responsible for all formatting and lingual errors present in this thesis.

I would like to thank my family and friends for supporting me through my thesis and especially my mother for her unconditional support throughout my life and for being the most motivating mathematics teacher. I would also like to thank my brothers for their support and humour and my boyfriend for his support and encouragement.

Petra Raussi

Espoo 18.12.2017

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

6LowPAN	IPv6 over Low-Power Personal Area Networks
ACSI	Abstract Communication Service Interface
API	Application Programming Interface
ASDU	Application Service Data Unit
AUMS	Active User Management System
BPL	Broadband Power Line
CCAPI	Control Centre Application Programming Interface
CDC	Common Data Class
CID	Configured IED Description
CIM	Common Information Model
CMS	Cluster Management System
CN	Control Node
COSEM	COmpanion Specification for Energy Metering
CPSA	Clustering Power Systems Approach
CPSM	Common Power System Model
CR	Cognitive Radio
CRN	Cognitive Radio Network
CT	Current Transformer
DCS	Distributed Control System
DER	Distributed Energy Resources
DERri	Distributed Energy Resources Research Infrastructure
DLMS	Device Language Message specification
DMS	Distribution Management System
DPWS	Devices Profile for Web Services
DSA	Dynamic Spectrum Access
DSL	Digital Subscriber Lines
DSO	Distribution System Operator
DSTP	DataSocket Transfer Protocol
EC	European Commission
EMS	Energy Management System

EPRI	Electric Power Research Institute
ERCOT	Electric Reliability Council of Texas
ERIGrid	European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out
EU	European Union
GOOSE	Generic Object Oriented Substation Events
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile Communications
GUI	Graphic User Interface
HAN	Home Area Network
HEMS	Home Energy Management System
HIL	Hardware-in-the-Loop
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
I/O	Input/Output
IAS	Intelligent Automation Services
ICD	IED Capability Description
ICT	Information and Communication Technology
ID	Intrusion Detection
IDE	Integrated Development Environment
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
IP address	Internet Protocol address
IRM	Interface Reference Model
ISO/OSI	International Standards Organization / Open System Interconnection
JaNDER	Joint Research Facility for Smart Energy Networks with Distributed Energy Resources
JRA	Joint Research Activity
JSON	JavaScript Object Notation
LAN	Local Area Network

LCP	Primary Control Logic
MAS	Multi-Agent Systems
MCU	Microcontroller
MG	Microgrid
MGC	Microgrid Community
MMS	Manufacturing Messaging Specification
MN	Managing Node
MPAM	EU-PC Double Degree Master Program in Automation/ Mechatronics
NA	Networking Activity
NAN/FAN	Neighbourhood Area Network/Field Area Network
NCC	Network Control Centre
NIST	National Institute of Standards and Technology
OPC	Object Linking and Embedding for Process Control
OSI	Open System Interconnect
OWL	Web Ontology Language
PC	Personal Computer
PLC	Programmable Logic Controller, Power Line Communication
PRP	Parallel Redundancy Protocol
PV	Photovoltaics
QVT	Query/View/Transformation
RDF	Resource Description Framework
RDFS	RDF Schema
RI	Research Infrastructure
RTDB	Real-Time DataBase
RTU	Remote Terminal Unit
RUA	Active User Router
SCADA	Supervisory Control and Data Acquisition
SCD	Substation Configuration Description
SCL	Substation Configuration Language
SDSL	Symmetric Digital Subscriber Line
SEP	Smart Energy Profile

SLD	Single Line Diagram
SMV	Sampled Measurement Value
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SOF	Start of Frame
SSD	System Specification Description
SVG	Scalable Vector Graphics
TA	Transnational access Activity
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TRILL	Transparent Interconnection of Lot of Links
UCA	Utility Communication Architecture
UDDI	Universal Discovery, Description and Integration
UDP	User Datagram Protocol
UML	Unified Modeling Language
UTC	Coordinated Universal Time
VPN	Virtual Private Network
VPP	Virtual Power Plant
WPAN	Wireless Personal Area Network
WSDL	Web Service Description Language
XMC	eXtensible Modelling and Control environment
XML	eXtensible Markup Language

1 INTRODUCTION

1.1 Background

The World and especially European Union (EU) are thriving towards renewable energy system because of the threats posed by anthropogenic global warming (IPCC 2014; Directive 2009/72/EC; Directive 2012/27/EU). Opposed to centralized energy resources applied by traditional power systems, distributed energy resources (DER) are cornerstones of a future renewable energy system, the smart grid. Smart grid can be defined multiple different ways and according to European Commission (EC) a smart grid is a power network, which integrates all actors, such as generators and consumers, cost-efficiently to provide security of supply, safety, high level of quality and low losses in a sustainable power system (COM(2011) 202, final, 2). Moreover, a smart grid also embodies Information and Communication Technology (ICT) to increase the intelligence of the power system, by which complex grids can be managed more efficiently and effectively (COM(2011) 202, final, 2). Figure 1.1 portrays the shift from traditional systems to smart grids.

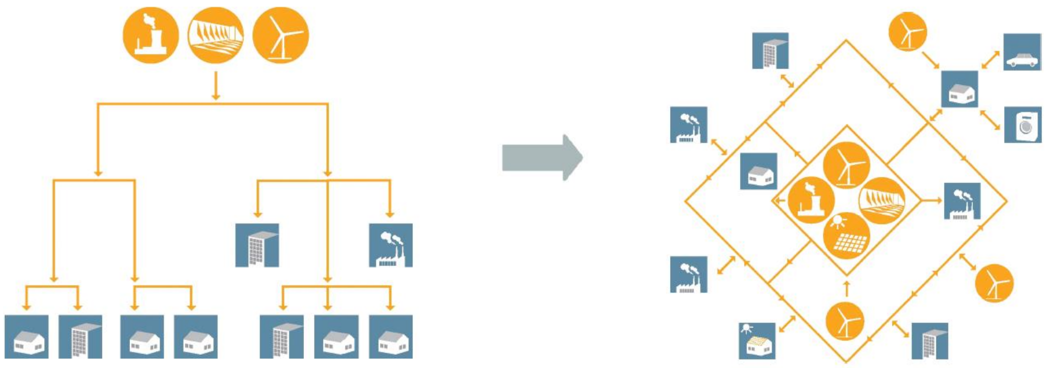

Figure 1.1. Diagrams of a traditional hierarchical power system and a network-structured smart grid (ABB 2008, 6).

Resulting from the decentralized nature of future energy production, new challenges for power systems have been realised. The new challenges have been caused by a shift in the political focus of integrating renewables to power systems and thriving consumers to actively participate in reducing their energy consumption via new tools (Ivanov 2016, 22). In order to maintain the level of power quality and reliability present in traditional power systems advanced automation and communication technologies, which were not previously required, are needed in smart grid systems (AIT et al. 2014, 2). Moreover, there are multiple data exchange standards already used by electrical power engineering industry. While these standards have previously been enough because of the lack of data exchange, now as the amount of data exchange is constantly increasing the utilization of all these multiple standards is creating complex interfaces and structures instead of the simple data exchange needed. As the electrical power systems continue to develop towards smart grids and the demand for data exchange within power systems accelerates, the data exchange standards and protocols must be integrated into a simple and flexible data exchange model (Ivanov 2016, 22).

Data exchange and communication play important role in the future smart grids. Previously, there was no need for major communication between the power system components, since the traditional power systems are heavily centralized. However, the smart grids are decentralized including the control and protection of the power system. Already it has been determined that combining traditional protection of power system and DER can have severe results since fault current comes from multiple directions in a power system with DER, which makes accurate detection and isolation of the fault immensely difficult and may cause tripping of small power plants or healthy power lines (Lakervi & Partanen 2008, 212). Therefore, communication between DER units and other components of the power system is crucial for the reliable operation of protection. Overall, DER has an effect on voltage control, bi-directional power flow, altered transient stability and raised fault levels, which could be mitigated by improved data exchange (Xu et al. 2011, 2088).

However, the main driver for moving towards smart grids and increasing the advanced automation and ICT level in the power systems is derived from the economy. Adaptation to the changes caused by decentralization could be conducted by implementing a range of other solutions, such as investing in passive grid components. Furthermore, as said these solutions often require more investments to the power system, which leads smart grids to be one of the best techno-economical solutions thanks to the monetary savings in comparison with other solutions for maintaining the current level of reliability and power quality, while the amount of DER increases. Therefore, the demand for advanced power system automation has grown and reliable high-quality communication technologies and protocols such as IEC 61850 need to be implemented. (Elgargouri, Elfituri & Elmusrati 2013, 1.)

1.2 ERIGrid project

EU has been one of the leading entities in areas of renewable energy systems and smart grids in research, regulation and implementation. Multiple smart grid projects conducted under EU have targeted different aspects of smart grids and one of these aspects is the communication technologies researched under European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out (ERIGrid) project (AIT et al. 2014, 2). Objectives of ERIGrid are to combine European smart grid research to the same platform, provide access to the state-of-art research facilities around Europe, integrate and unify research on smart grid configurations and develop the sustainable integration of DER (AIT et al. 2014, 5–6). ERIGrid is conducted to increase the efficiency of electric power engineering laboratories, to promote research result transfer within research facilities and industry and to enable research facilities to extend research possibilities by collaboration (AIT et al. 2014, 5–6). ERIGrid is a 54 month-long project, which is divided into coordination, networking activities, joint research activities and transnational access activities, presented in Figure 1.2. Network Activities (NA) aim to improve the trans-national access possibilities provided for external users. Joint Research Activities (JRA) intend to integrate the research infrastructures to allow better services offered. Transnational access Activities (TA) are meant for organizing the calls for providing access to the research facilities. (AIT et al. 2014, 32–34.)

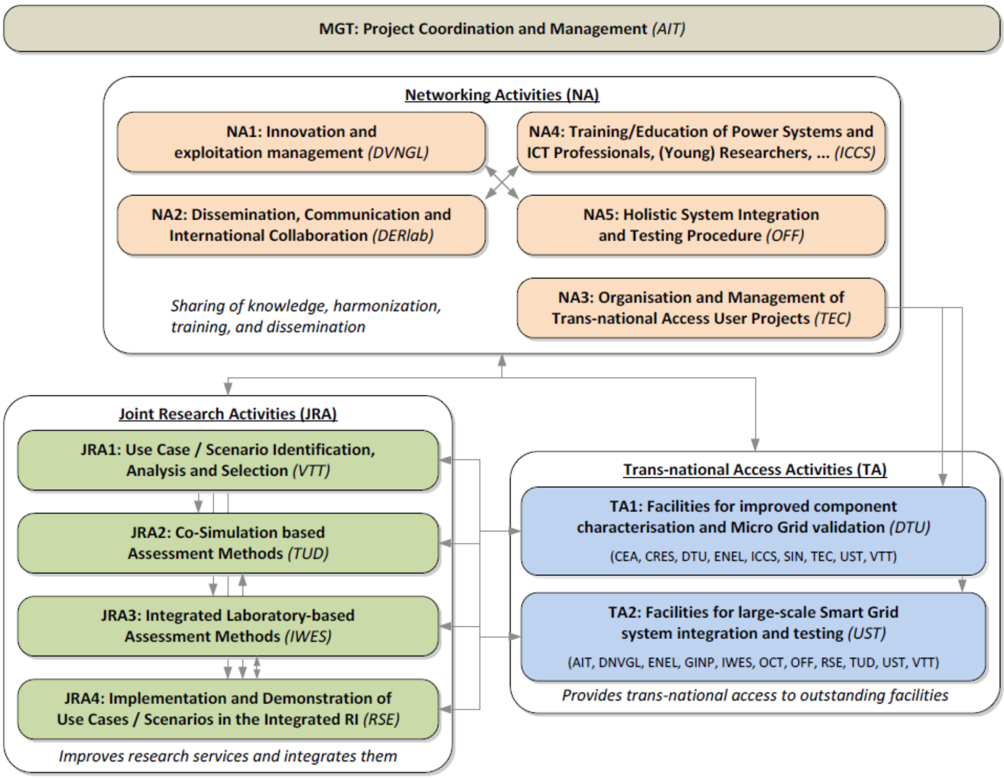

Figure 1.2. Integrating activities and work packages (AIT et al. 2014, 33).

The project is separated into 12 work packages, which are further divided into tasks. This thesis is mainly conducted under work packages JRA3 Integrated Laboratory-based Assessment Methods and JRA4 Implementation and Demonstration of Use Cases / Scenarios in the Integrated Research Infrastructure (RI). The tasks within work packages JRA3 and JRA4 are JRA3.1-JRA3.4 and JRA4.1-JRA4.3. JRA3 tasks are for improving smart grid ICT, real-time simulation and HIL (Hardware-in-the-Loop) methods, system integration and testing methods. JRA4.1 aims to improve JaNDER (Joint Research Facility for Smart Energy Networks with Distributed Energy Resources) at each participating research infrastructure to interconnect laboratories. JRA4.2 intends to demonstrate and validate the integration of laboratories, first as the integration of single RI and then of multiple RIs. JRA4.3 is conducted to analyse and evaluate the results of the implementation of tasks JRA4.1 and JRA4.2. (AIT et al. 2014, 53–57.)

1.3 Research problem, objectives and delimitation

This thesis is written for VTT Technical Research Centre of Finland regarding the implementation of real-time laboratory interconnection proposed in the ERIGrid project. The purpose of this thesis is to implement a real-time laboratory interconnection to allow a remote access to VTT's Multipower laboratory for other partner laboratories in the ERIGrid project. Additionally, the interconnection can be accessed locally by VTT and visitors of VTT for testing and researching. The local access to the Multipower laboratory is specially granted through TA for those visitors, who do not have own electrical power system laboratory.

This thesis provides answers to the following research questions:

- How does an interconnection between laboratories operate?
- What kind of interconnections already exist?
- How could this particular laboratory interconnection be implemented?
- In which cases interconnection would be beneficial and to who?
- Which laboratories and facilities this interconnection could be applied to?
- How long can be an example delay between information sent and received?

This thesis focuses on implementing the real-time interconnection at VTT's Multipower laboratory. Implementation of the real-time interconnection at other ERIGrid partner laboratories is out of the scope of this thesis. Additionally, the interconnection implemented in this thesis is limited to implementation of a local real-time database replication connection to a cloud-based real-time database, though in the scope of the ERIGrid project the implementation is also conducted as an IEC 61850 interface and a Common Information Model (CIM) interface.

1.4 Research methodology

This thesis is conducted as an implementation of the real-time laboratory interconnection, which will include installing the required programs, programming the interconnection, establishing test cases for the real-time interconnection and conducting the testing to confirm proper operation and reliability of the real-time interconnection. The thesis also includes literature review discussing state-of-art research, relevant standards and protocols, and VTT's Multipower laboratory.

The main challenge of this thesis is establishing the connection between the local real-time database and the electrical process for data transfer. Additional, challenges could occur regarding installation and configuration of the programs needed for the communication gateway and regarding the testing if there is no partner laboratory available at a suitable time for the test cases.

1.5 Structure of the thesis

This thesis is divided into 8 chapters, of which the first chapter is an introduction. The second chapter covers literature review and discusses similar projects conducted previously. A general overview of IEC 61850 standard and CIM are discussed in the third chapter, while also providing insight into details relevant for the implementation of a real-time laboratory interconnection. The fourth chapter discusses the Multipower laboratory facility at VTT and the fifth chapter discusses JaNDER. The sixth chapter describes the implementation process of the real-time laboratory interconnection at Multipower laboratory and includes sections regarding installations, programming and testing of the interconnection. The seventh chapter discusses the results of the interconnection testing and overall implementation of the real-time laboratory interconnection. Conclusions are covered by the eighth chapter.

2 LITERATURE REVIEW

This chapter discusses the research and projects conducted previously regarding laboratory interconnections in the field of smart grids. There are multiple possibilities to implement interconnection and different types of interconnections should be considered, as there are interconnections between two or multiple laboratories, between laboratory and laboratory resource in distance from the core facility within electrical power systems and between smart grid resources. The previous research on laboratory interconnections is discussed to establish a background for the implementation conducted in this thesis. Further, this section provides multiple examples of implementation possibilities and communication technologies, which provides the basis for the discussion and comparisons between the previous implementations and the implementation in this thesis.

Chapter 2 is divided into seven subsections. Section 2.1. discusses previous research on laboratory remote connections, which allow accessing the laboratory equipment from a remote computer. Section 2.2 provides examples of several different options for communication implementation for DER remote connections and interconnections. Section 2.3 provides examples of different communication technologies, which could be used in the implementation of smart meter ICT on different levels of the power system. Section 2.4 discusses connection implementations based on redesigned power grid possibilities. Section 2.5 discusses previously implemented laboratory interconnections. Section 2.6 discusses mapping of the ICT communication examples discussed in Sections 2.1–2.5. Summary of the discussion and examples in the Sections 2.1–2.6 is provided in the Section 2.1.7 alongside with discussion on lessons learned from the development of the interconnection implementation.

2.1 Laboratory remote connection

Remote connections have been mainly an interest of remote laboratory education for years, as remote connection allows distance learning with broader possibilities. For instance, MPAM (EU-PC Double Degree Master Program in Automation/Mechatronics) project funded by European Commission was conducted to enable distance learning of MCUs (microcontroller) by remote connection to a laboratory stand at Saint-Petersburg State Electrotechnical University “LETI” (Belgradskaya & Ignatiev 2014, 11).

Multiple laboratories’ remote connections have been implemented mainly for educational purposes. A remote connection can be implemented several different ways. In the examples, the implementations were conducted by using WebSocket protocol on Transmission Control Protocol (TCP) connection via Hypertext Transfer Protocol (HTTP), or entirely as Transmission Control Protocol/Internet Protocol (TCP/IP) connection via Ethernet Local Area Network (LAN), or by EJS Applets and TwinCAT connection via the Internet, or by synchronous serial communication automatically connecting via the Internet (Sáenz et al. 2016, 143–144; Hegedus et al. 2013, 257; Mostefaoui, Benachenhou & Benattia 2017, 481–484; Besada-Portas, Lopez-Orozco, De La Torre & De La Cruz 2013, 156–158; Sendoya-Losada, Silva & Waltero, 2016, 11503–11504, 11511). The electronics and mechatronics laboratory remote connections mostly follow the architecture of connecting a remote Personal Computer (PC) to a laboratory server via the Internet. In Sáenz et al. (2016, 143–144) a remote connection was implemented with WebSocket protocol, which uses JavaScript Object Notation (JSON) data format. WebSocket enables two-way communication between client and server via TCP connection by utilizing HTTP technologies and infrastructures (Sáenz et al. 2016, 143–144).

In Hegedus et al. (2013, 257), the laboratory remote connection was conducted entirely as TCP/IP. The connection between client and server was implemented via Ethernet LAN connected to Virtual Private Network (VPN), while the data transfer was done by running an Apache HTTP server (Hegedus et al. 2013, 257). Similarly, Mostefaoui et al. (2017, 481–484) implemented a remote connection via Ethernet LAN and TCP/IP utilizing socket Access Method with Arduino board. In this case, the data transfers were also conducted as HTTP requests (Mostefaoui et al. 2017, 481–484).

In Besada-Portas et al. (2013, 156–158), a remote connection was conducted with EJS (Easy Java Simulations) Applets and TwinCAT. EJS Applets communicate with Programmable Logic Controllers (PLC) via Java server application and the data transfer was conducted with TCP sockets and TwinCAT data communication library (Besada-Portas et al. 2013, 156–158). Sendoya-Losada et al. (2016, 11503–11504, 11511) implemented control and automation training module remote connection by creating an automatic wiring system on Arduino UNO board, which could be controlled by synchronous serial communication and automatically connected via the Internet.

2.2 DER remote connections and interconnections

Within the smart grid sphere, possibilities regarding remote connection have been realised quite recently and research on smart grid remote connections for small DER units has not yet been conducted in huge numbers. Therefore, the number of previously conducted projects is somewhat modest. However, the remote DER connections have been used for larger DER units, for instance, wind turbines by PLC via Ethernet TCP/IP LAN or WLAN communication to Supervisory Control and Data Acquisition (SCADA), already for decades (Mäkelä 2016, 31–33). However, increasing amount of research is conducted to better understand and implement ICT on smart grids. A major part of this research are DER remote connections and interconnections, since it has been realized that the protection of power systems must be expanded with ICT communication as in fault situation DER may cause inaccuracies because of a fault current coming from multiple directions, which may cause tripping of small power plants or healthy power lines. Without implementing ICT communications and advanced automation, the growth of DER units in the power system is limited by sustained over-voltage caused by electricity production in weak grids and by increasing fault current or overloading components on the normal operational state in stiff grids (Lakervi et al. 2008, 211–212).

There are multiple ways to implement DER remote connection or interconnection, of which the most relevant research for this thesis is discussed in the following. In the examples, the DER communication connections were established with CIM based POWERLINK protocol connection between Energy Management System (EMS) and DER inverters, or with IEC 61850 based WebSocket protocol connection as iPower Flexibility Interface via cloud service, or with IEC 60870-5-104 protocol and TCP/IP connection via LAN or WAN, or with combination of protocols such as Transparent Interconnection of Lot of Links (TRILL) and RBridge in an Ethernet legacy system or high-rate wireless communications system and Broadband Power Line Communication (BPL) (Wlas, Gackowski & Kolbusz 2011, 2070–2075; Orda et al. 2013, 314–316; Kolenc et al. 2017, 47–48, 50; Selga, Zaballos & Navarro 2013, 588–591; Della Giustina, Repo, Zanini & Cremaschini 2011, 102–105; Della Giustina, Andersson & Ravera Iglesias 2013, 1–2).

In Wlas et al. (2011, 2070–2075), the DER integration was implemented via POWERLINK protocol. The implementation was based on CIM and the communications included three communication technologies for communication between EMS and inverter of DER, which were Ethernet, POWERLINK and IEC 61850. There are multiple other protocols, which could have been utilized for the integration, including Modbus/TCP, Ethernet/IP, SERCOS III, EtherCAT and Profinet. The chosen protocol Ethernet POWERLINK is based on the IEEE 802.3 mechanisms and standard featured in CANopen. The protocol contains Managing Nodes (MN) and Control Nodes (CN). MN manages the network and CN operates as a slave. POWERLINK transfers the data in cycles of transferring, configuring and waiting for the next cycle. The Ethernet POWERLINK technology used could be implemented on fibre-optic cable to allow possibly greater distances between the devices in the power system, which is common in the case of DER. (Wlas et al. 2011, 2070–2075.)

Orda et al. (2013, 314–316) implemented an IEC 61850 based DER interconnection through iPower Flexibility Interface, which is targeted towards direct control of DER and refers to two-way communication between DER and aggregator. Each DER communicates its local flexibility to the aggregator, according to which DERs are controlled by aggregators. The DER connection was implemented via cloud service, which handled the connections between DERs and network and stores data. Real-time communication between DER and cloud was based on WebSocket protocol. DER transferred the raw data via WebSocket to the cloud service and DER controllers provided custom modules to modify the raw data into data structures, which could be utilized by the cloud service. The cloud service was implemented as a multi-tenant cloud service using Microsoft Azure3. The implemented cloud service supported communication with Virtual Power Plants (VPP), SCADA and client monitor applications. The communication was conducted via eXtensible Markup Language (XML) web service and a web-based RESTful interface as presented in Figure 2.1. (Orda et al. 2013, 314–316.)

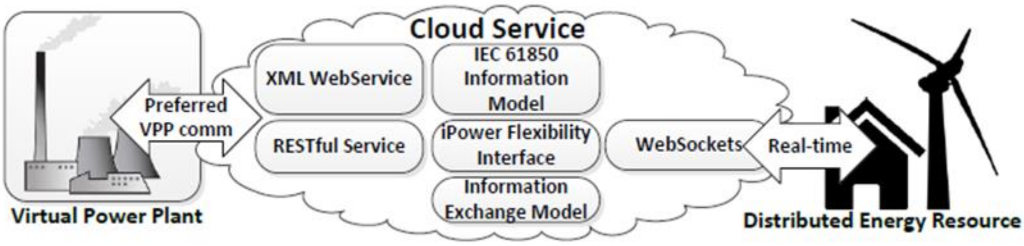

Figure 2.1. An example VPP and DER communication via a cloud (Orda et al. 2013, 316).

In Figure 2.1, there are two possible interfaces for the communication, both based on IEC 61850. The XML web service returns a response via Simple Object Access Protocol (SOAP) XML and RESTful interface via plain XML or JSON, which the client must declare in the HTTP request. With the RESTful interface, each DER can be accessed and updated via URL nodes and the objects are named following the naming structure of IEC 61850 standard. (Orda et al. 2013, 315–316.)

Kolenc et al. (2017, 47–48, 50) implemented a DER remote connection to a VPP based on TCP/IP and IEC 60870-5-104 protocol. Each DER had a Remote Terminal Unit (RTU), through which communication module the DER was connected to the rest of the communication system. A part of this implementation was a leased Symmetric Digital Subscriber Line (SDSL), which was used to secure communication via the remote connection. TCP/IP interface implementation also allowed a connection to the LAN or WAN since this expands the distance limit between DER and local network. (Kolenc et al. 2017, 47–48, 50.)

In Selga et al. (2013, 588–591) the implementation of the DER remote connection was conducted with a combination of protocols. The combination was Transparent Interconnection of Lot of Links (TRILL) and RBridge as these protocols support Ethernet switches, which is important because of the high number of Ethernet devices in the system. In this implementation, the initial Ethernet frame was encased with an Ethernet header by RBridge. Then a TRILL header was entered between the Ethernet frames with the RBridge identification of the last RBridge to the frame. This method allowed frames to be transferred between RBridges on larger networks than Ethernet while being more secure. On the final RBridge, the TRILL and Ethernet headers were removed and the initial Ethernet frame was transferred forward. Figure 2.2 displays the entire messaging process for legacy Ethernet devices. This method of combining protocols has improved the security of data transfer. However, the combination of TRILL and RBridge protocols lacked resiliency necessary for smart grid as a result of which RBridge protocol should be substituted with Parallel Redundancy Protocol (PRP) protocol to overcome resiliency issues. (Selga et al. 2013, 588–591.)

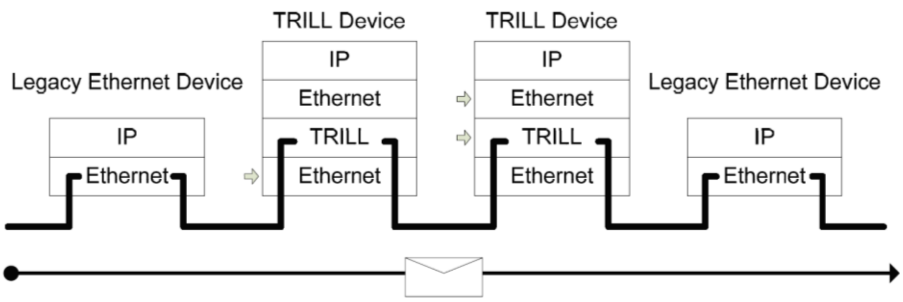

Figure 2.2. The protocol stack for legacy Ethernet devices (Selga et al. 2013, 590).

Della Giustina et al. (2011, 102–105) also used a combination of protocols in the implementation by combining hybrid broadband power line communication (BPL) and a high-rate wireless communications system, which could be point-to-point Wi-Fi because of the low implementation costs and free license. In this implementation, control centre was connected with Primary and Secondary Substations via a combination of BPL over MV cables, Fibre Optics (FO) ring and Wi-Fi links as presented in Figure 2.3. (Della Giustina et al. 2011, 102–105; Della Giustina et al. 2013, 1–2.)

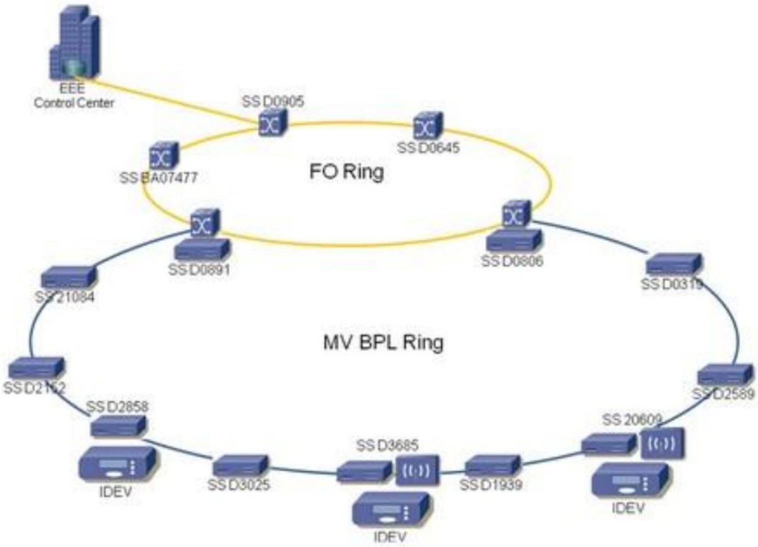

Figure 2.3. Diagram of the communication architecture with BPL over MV cables, FO ring and Wi-Fi links. (Della Giustina et al. 2013, 2).

BPL and Wi-Fi were combined with the aid of a routing algorithm on the International Standards Organization / Open System Interconnection (ISO/OSI) stack layer 2. Since routing was conducted at layer 2, everything connected to the same feeder could be considered as parts of the same LAN and IEC 61850 standard defines Generic Object Oriented Substation Events (GOOSE) messages for real-time communication also on the layer 2, therefore GOOSE messages are conveniently supported by the combination. For more details on the ISO/OSI stack model, see the Section 2.6. The remote connections could be also synchronized by connecting the LAN clock to Global Positioning System (GPS) and by using SNTP protocol to distribute the LAN clock information. (Della Giustina et al. 2011, 102–105.)

2.3 Smart meter ICT communication technologies

There are not only connection possibilities for connecting remote DER units, but also interconnections within the smart grid on multiple communication levels. In this section, a general overview of ICT communication is provided, while also the relevant smart meter ICT communication technologies, ZigBee, Mesh network, Power Line Communication (PLC), Digital Subscriber Lines (DSL), cellular networks, Cognitive Radios (CR) and IPv6 over Low-Power Personal Area Networks (6LowPAN), are discussed (Güngör et al. 2011, 530–534; Islam, Mahmud & Oo 2016, 128; Khan, Rehmani & Reisslein 2016, 860–863). Additionally, examples of smart meter ICT communication technologies, CR, Wirepas and 6LowPAN, will be provided (Khan et al. 2016, 861–862; Wirepas 2017, 3–5; Jacobsen & Mikkelsen 2014, 131–133).

According to Güngör et al. (2011, 530) and Islam et al. (2016, 128), the smart meter communication can be considered to have at least two levels of communication, the first level being communication between appliances and smart meters and the second level being communication from smart meters to Distribution System Operators (DSOs) and vice versa. The communication can be wired or wireless, which both have advantages as wireless technologies have lower costs and can reach remote areas, while wired technologies do not depend on batteries or face interference issues. There are several suitable technologies for wireless and wired communication. For the first level of communication, between appliances and smart meters PLC and DSL or wireless communication, for instance, ZigBee, 6LowPAN, Z-wave or mesh network, could be used. Similarly, computer networks, for instance, the Internet and cellular networks could be used for the communication between smart meters and DSOs. (Güngör et al. 2011, 530; Islam et al. 2016, 128.)

According to Khan et al. (2016, 863), the smart meter network can be divided into three layers, which are displayed in Figure 2.4. Home Area Network (HAN) is the network within homes and connects smart appliances with smart meters. Neighbourhood Area Network/ Field Area Network (NAN/FAN) is a combination of multiple networks within homes and connects the smart meters to DSO's connection point or a network gateway, such as communications tower, utility pole-mounted device or power substation. Via NAN, the data from smart meters and HANs is collected and transmitted further to the Wide Area Network (WAN). All the gathered data are transmitted via WAN to the substations and utility data centres. (Khan et al. 2016, 863.)

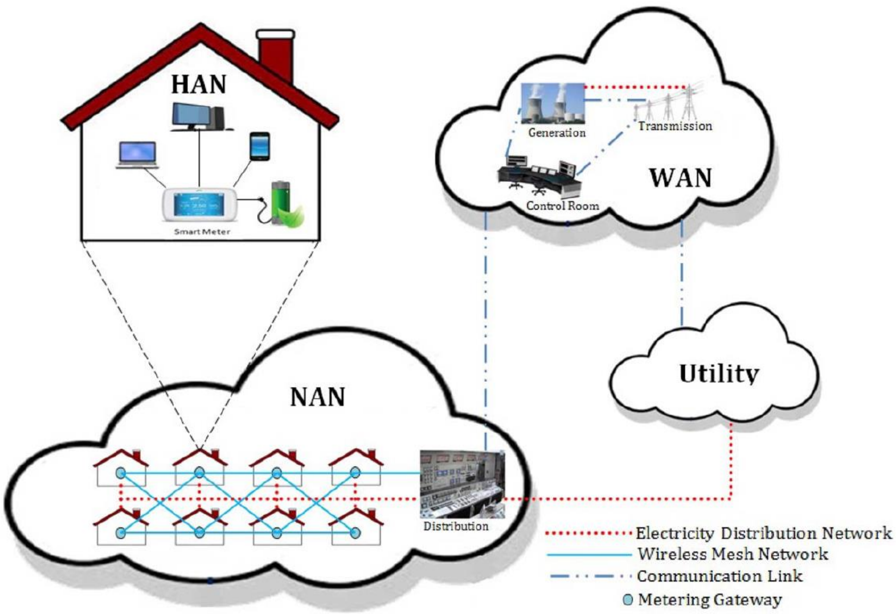

Figure 2.4. Smart grid architecture as a combination of Home Area Network (HAN), Neighbourhood Area Network (NAN) and Wide Area Network (WAN) (Khan et al. 2016, 860).

ZigBee can be implemented especially for smart lighting, automatic meter reading, energy monitoring and home automation. ZigBee enables the communication and control of smart meters and appliances utilizing it. ZigBee has also Smart Energy Profile (SEP) application, which allows the ZigBee system to send messages to the consumers and provides data on their real-time consumption to the consumers. Implementation with ZigBee can be beneficial since ZigBee has low bandwidth requirements and costs while operating without a licence and being standardized on IEEE 802.15.4 standard. However, as wireless communication technology ZigBee may be vulnerable to interference, which could increase the opportunities for corrupting the entire smart home system. (Güngör et al. 2011, 530–531.)

Mesh network is another wireless option for smart meter communication. In a mesh network, the network is implemented as a group of nodes, which act as independent routers. The node implementation allows new nodes to be added to the mesh network. While the self-healing properties allow communication signals to identify new routings through active nodes if nodes previously used drop from the mesh network. In a mesh network implementation, all smart appliances have radio module, via which they can route the communication signals, while smart meters have a signal repeater for transferring the data in the network until it reaches the correct access point. The main benefits of the mesh network are the self-healing and self-organizing properties with low costs. However, the major issue of mesh networks is the smart application and meter density as a low density of routers and signal repeaters can cause fading or lack of capacity in the network. Additionally, the signals might have to be encrypted to avoid security issues, since the communication signals pass through all the access points in the mesh network. (Güngör et al. 2011, 531; Islam et al. 2016, 128.)

PLC transfers high-speed data signals between appliances by using the existing power lines. Usually, the power lines are used to connect smart meters to data concentrators while the communication between data concentrators and data centres is implemented with cellular network technologies. Using existing power lines reduces the deployment costs, yet the disturbances and noise in the power lines can cause issues with signal quality. Therefore, PLC has low bandwidth, which limits utilization of PLC with applications requiring higher bandwidth. However, PLC supports HAN and combined with other technologies, for instance, General Packet Radio Service (GPRS) or Global System for Mobile Communications (GSM), reduces the disturbance and noise issues. (Güngör et al. 2011, 533–534; Khan et al. 2016, 861.)

Digital Subscriber Lines (DSL) are implemented similarly to PLC, only using voice telephone network wires. DSL also transfers high-speed data signals and the main advantages lay in the reduced deployment costs resulting from the utilization of telephone network. Contrary to PLC, DSL allows usage of high bandwidth data transfer. However, DSL implementations might have issues in rural areas as the network is sparse and the DSL quality is based on the distance between the customer and serving telephone exchange. Hence, DSL implementation might not be possible for mission-critical applications. (Güngör et al. 2011, 534.)

Cellular network can be used in the communication between a smart meter and DSO. Available cellular network technologies include 2G, 2.5G, 3G, LTE and WiMAX. The advantage of cellular networks is that smart meter communication can be added to the already existing network, which decreases the deployment costs and increases the coverage of the cellular network. Cellular networks also support multiple different communication systems and protocols, for instance, GSM supports AMI and HAN applications. In addition, most of the security issues have already been solved when establishing the technology, although congestions and reduced network performance could lead to lack of reliability in emergencies. As a result of these reliability issues, DSOs are more interested in building a private cellular network rather than using the cellular network shared with customer market. (Güngör et al. 2011, 531–532.)

According to Khan et al. (2016, 861–862), there could be issues regarding lack of capacity in the spectrum caused by inefficient utilization. The smart meter communication on cognitive radios (CR) could use these inefficient sections of the spectrum. CR is based on Dynamic Spectrum Access (DSA). CR utilizes the existing spectrum for transmitting signals. The idea of CR is to access licensed bands opportunistically while avoiding interference of the licensed users. CRs seek white space or spectrum holes, which is the vacant portion of the spectrum, and then use the best available channel found for transmitting signals, while licensed users are not using the channel. Additionally, spectrum sharing can be used temporally to improve the CR communication. CR implementation can be built into a Cognitive Radio Network (CRN), which could cover all the CR devices in the smart grid. (Khan et al. 2016, 861–862.)

Similar to CR, there are several other communication technologies available based on the Wireless Sensor Networks (WSN). An example of a WSN based technology is Wirepas. Wirepas is a decentralized wireless Internet-of-Things (IoT) communication protocol. Each device containing radio chip with Wirepas includes intelligence for local decision-making and co-operation. Therefore, Wirepas has no need for central control or external intervention. Similar to Mesh network, Wirepas has self-healing properties, which allow Wirepas to reroute via active nodes if a node used drops from the network. Wirepas can also reroute similarly to avoid obstacles in between nodes. The benefits of Wirepas include large coverage, high reliability and low latency. (Wirepas 2017, 3–5.)

In a smart home system example by Jacobsen et al. (2014, 131–133), there are multiple smart devices, which are connected to the ICT services or to one another. The smart home system could be for instance Home Energy Management System (HEMS), which monitors and controls the energy consumption of the house and can be operated to avoid the peaks in the energy prices on the market. Systems as HEMS have benefits for both consumers and the DSO. In HEMS infrastructure proposed by Jacobsen et al. (2014, 131–133), the residential house includes smart devices and their controls, while receiving management platform and Intelligent Automation Services (IAS) from a cloud infrastructure. The implementations can be built on IPv4, however globally there is a shortage of IPv4 addresses, which encourages home systems to have IPv6 based deployment, for instance, IPv6 over Low-Power Personal Area Networks (6LowPAN). 6LowPAN allows HAN connections via open protocol layer, which can be more convenient than using, for instance, IEEE 802.15.4 radio MAC of ZigBee version 1 and 2. Newer versions of ZigBee and HomePlug have also adapted by providing protocols for HAN. IPv6 can also adapt to Wireless Personal Area Network (WPAN), Ethernet and Wi-Fi. Smart devices can transfer information via radio communication also, which is mainly used in communication between smart devices within the smart home. (Jacobsen et al. 2014, 131–133.)

2.4 Redesigned power system connections

Currently, the remote connection and interconnection and smart grid ICT communication technologies possibilities have been expanded to the research of different kinds of clustering, grouping and connecting possibilities of smart grids, DERs, microgrids and other sections of grids. In the examples, information will be provided on microgrid communities, Clustering Power Systems Approach (CPSA) and Web-of-Cells (Tian et al. 2017, 56–61; Leksawat et al. 2015, 1–4; Martini et al. 2015, 2–3; Merino et al. 2015, 2–3).

According to Tian et al. (2017, 56–58), a smart grid interconnection could be understood as a microgrid community (MGC), where multiple microgrids (MG), small generation and distribution systems, are connected as a community in a small geographical area to increase the reliability of their power supply. MGC should also be capable of operating in islanding mode. MGC has more layers in the grid management and control than a microgrid, which is directly connected to the DER. The controls of MGC could be implemented based on a master-slave model or peer-to-peer model. In the master-slave model, one of the MGs is the master, while rest of the MGs remain as slaves, which adopt the voltage and frequency of the master. In peer-to-peer model, the MGs collectively support the voltage and frequency within the community, which increases the stability of decentralized systems. (Tian et al. 2017, 56–58.)

MGC was implemented based on IEC 61850 standard. MGC has had a data transmission module and used communication software and data acquisition for data transfer. The MGC control system has had a separate communication system to ensure the reliability of controls. (Tian et al. 2017, 58, 61.)

Leksawat et al. (2015, 1–3) implemented a smart grid interconnection by combining DERs and other electrical elements in the power systems into cluster networks based on Clustering Power Systems Approach (CPSA). The cluster networks could be implemented on transmission, distribution, local area and even unit levels. With this type of hierarchical structure both communication within a single cluster and between several clusters had to be considered. The clusters were controlled by cluster management system (CMS). CMS managed the communication and interoperation capabilities of a cluster. All the communication within the cluster networks was transferred via the Internet with VPN technology security. (Leksawat et al. 2015, 1–3.)

The communication between clusters was conducted with web service, which was based on service-oriented architecture (SOA) concept. Data was transferred via HTTP in XML format. Web service combines service provider, service requester and service registry and utilizes three technologies of SOAP, Web Service Description Language (WSDL) and Universal Discovery, Description and Integration (UDDI). Figure 2.5 displays the architecture of the combination of technologies. (Leksawat et al. 2015, 3–4.)

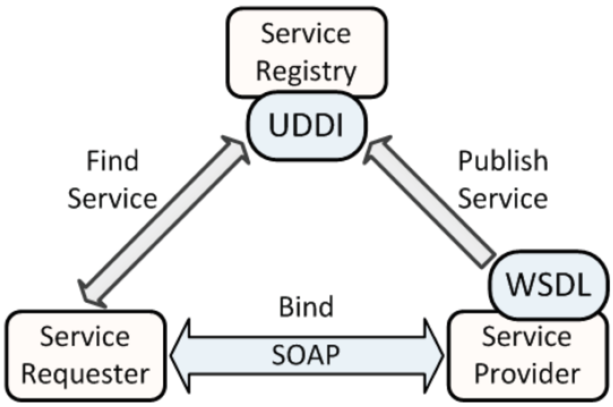

Figure 2.5. Diagram of technologies used in web service architecture (Leksawat et al. 2015, 4).

In the web service, information is transferred between service requester and service provider via SOAP messages. SOAP messages can include envelope, header, body and fault. SOAP envelope is the root element, and along with the body, an obligatory element of a SOAP message. WSDL describes service interface and service implementation descriptions, which service provider prepares. WSDL descriptions contain information about the available service name, guidance for usage and service location. UDDI is the service registry, which provides a framework for publishing and discovering web service descriptions. Moreover, WSDL publishes their services on UDDI, while service requesters find these available services from UDDI and can according to the guidance in the service description contact WSDL to conduct the service. (Leksawat et al. 2015, 3–4.)

In Martini et al. (2015, 2–3) and Merino et al. (2015, 2–3), a new architecture concept for smart grid interconnection was presented. In this implementation, the power grid was divided into cells. The cell is a section of the grid, which includes DER, loads and storage units. The cell also has clear boundaries, which are defined based on the physical grid and geographical area, and should have reserves to fix cell balancing and voltage issues locally. However, the cell is not expected to be able to operate in island mode. Together the cells form a web-of-cells and are connected via one or multiple inter-cell physical tie lines. (Martini et al. 2015, 2; Merino et al. 2015, 2–3.) Figure 2.6 shows an example of the web-of-cells concept.

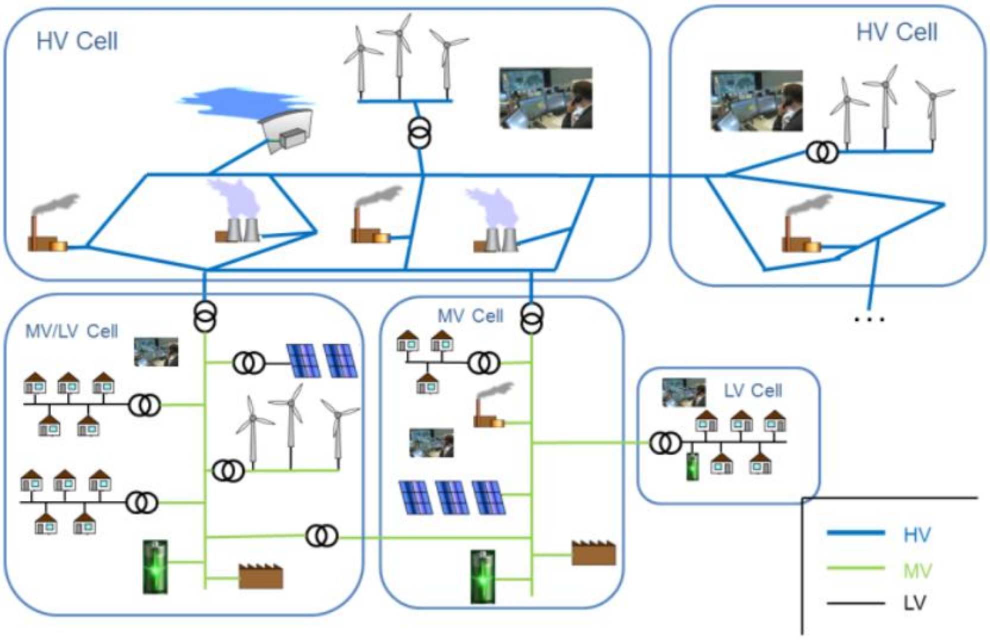

Figure 2.6. An example of web-of-cells concept (Martini et al. 2015, 2).

As presented in Figure 2.6, the cells can be of different voltages levels. The cells have a Control Cell Operator (CCO), which is similar to Transmission System Operator (TSO) and has the responsibility for local voltage and balance control. The web-of-cells concept concentrated on balance control rather than frequency control. (Martini et al. 2015, 2–3; Merino et al. 2015, 3.) According to Martini et al. (2015, 3), using balancing control increased safety in local imbalance situation and the balance optimization could be conducted with fewer reserves or reserves of other cells could be employed by a more economical operation. According to Merino et al. (2015, 3), the benefits of this concept were the concentration on solving local issues locally, which reduced the complexity and communication overhead.

2.5 Laboratory interconnection

Laboratory interconnection is used to connect two or multiple power system laboratories and their equipment together, similarly to the interconnections within smart grids. Laboratory interconnections are mainly implemented for educational or research purposes. Laboratory interconnections can connect microgrids or DER units to the laboratory facilities for testing, monitoring and control. In the examples, laboratory interconnections have been implemented by interconnecting external microgrids with a microgrid in the laboratory facilities, or by connecting laboratory microgrid to medium voltage grid of DSO, or by creating microgrid platforms of each laboratories' facilities, which have been connected to a common cloud-based SCADA, or by implementing a gateway interface, which has connected all laboratories to a joint SCADA (Messinis et al. 2014, 31–34; Sandroni et al. 2016, 1–3; Nguyen, Tran & Besanger 2016, 28–31; DERri 2013a; DERlab e.V. 2013, 26; DERri 2013b; CORDIS 2014, 9).

In Messinis et al. (2014, 31), the laboratory interconnection has included a couple of external microgrids, which have been connected to a microgrid within laboratory facilities. The connected microgrids could be both single phase and three phase grids. The microgrids could have included multiple smart grid applications and DER units, such as small wind turbines, photovoltaics (PVs), loads and batteries. Monitoring and controlling of the entire microgrids system have been conducted by SCADA via HMI (Human Machine Interface). (Messinis et al. 2014, 31.)

The communications have been implemented with intelligent load controllers and industrial communication protocols. The load controllers have multiple analog and digital Input/Output (I/O) and TCP/IP connectivity. The load controllers are able to obtain current, voltage and power consumption measurements of loads from two load switches, which each load controller drives. The load controllers are embedded systems, which support JADE and run Java Virtual Machine. JADE is a FIPA compliant platform based on Java, which allows the development of multi-agent systems (MAS). Industrial communication protocols, for instance, Object Linking and Embedding for Process Control (OPC), have been used for monitoring and controlling of DER. Within the single-phase microgrid, SCADA has been implemented by PLC programmed to CoDesys software as displayed in Figure 2.7. Controlling, monitoring and data acquisition has been conducted via LabVIEW running on the central PC of the microgrid. Microgrid AC side has been monitored by power meters, which use Modbus via RS-485 in the data transmission to the microgrid central PC, in which LabVIEW acquires measurements from the power meters and stores them. (Messinis et al. 2014, 32–33.)

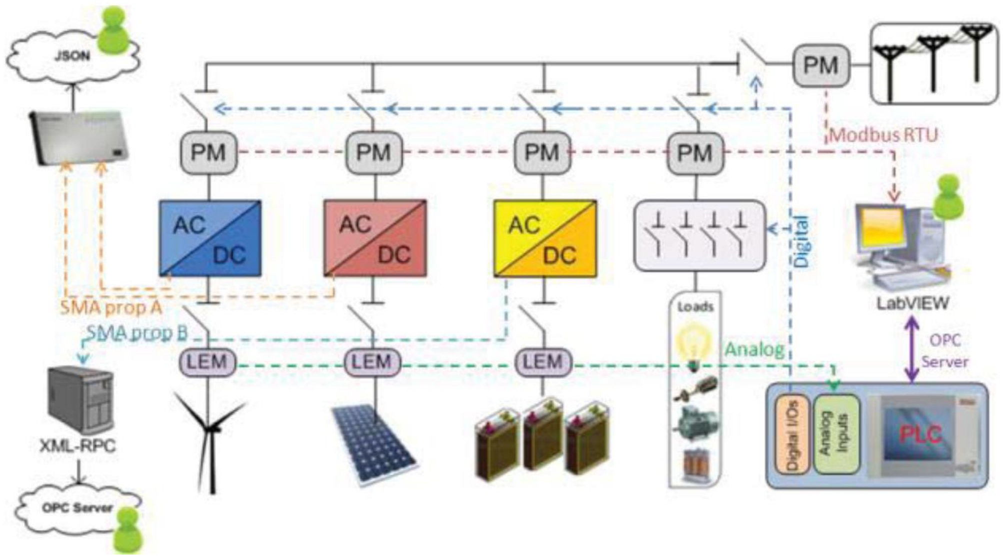

Figure 2.7. The infrastructure of the single-phase microgrid (Messinis et al. 2014, 34).

In Sandroni et al. (2016, 1–2), the laboratory interconnection has connected microgrid in the laboratory facilities to a medium voltage grid of DSO. The microgrid has contained multiple DER and smart grid applications, and a separate low voltage direct current (LVDC) microgrid with storages, loads and PV simulator. For monitoring and controlling, the laboratory has used its own SCADA. All the measurements from meters, grid analyzers and local controllers of the devices have been collected to SCADA, in which the data are used for user or control functions and stored for historical analysis. Control commands and set points have also been transmitted through SCADA. As the laboratory has developed own SCADA, it has been implemented on LabVIEW and allowed communication with several protocols, for instance, Modbus TCP, OPC, HTTP Web-services, CAN bus and RS-485. There has been also local control stations in the microgrid, which conduct the low-level communication and local control functions. High-level control functions use LabVIEW DataSocket Transfer Protocol (DSTP) for communication with SCADA via OPC Data Access, Modbus TCP or Redis gateway. (Sandroni et al. 2016, 1–2.)

The communication infrastructure between laboratory facilities and DSO has been implemented as internet-based public infrastructure on ADSL or optical fibre links. The communication has been based on IEC 61850 standard. The connection between DSO and laboratory facility has been implemented as the connection between DSO and active user. For active user implementation, Active User Router (RUA) has been set-up between laboratory microgrid and DSO as portrayed in Figure 2.8. RUA has been used to manage the information exchange between Active User Management System (AUMS) and Primary Control Logic (LCP), which is on the premises of the DSO at the Primary Substation. RUA uses IEC 61850 protocol to communication with LCP and Modbus TCP with AUMS. RUA transfer messages between LCP and AUMS by the sending side registering the message at the RUA Modbus holding registers and the receiving side either reading the messages or sending them further to LCP as IEC 61850 Manufacturing Messaging Specification (MMS) messages. (Sandroni et al. 2016, 3.)

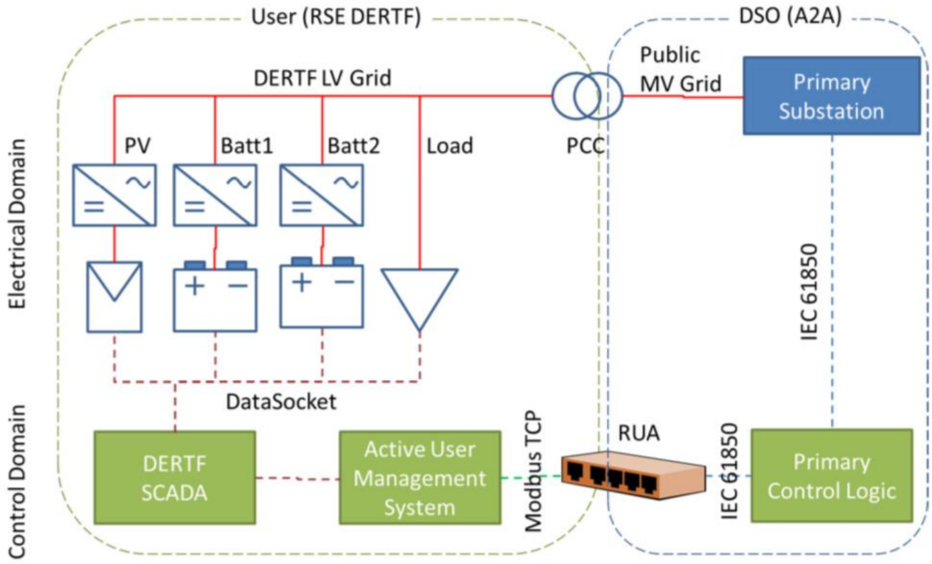

Figure 2.8. Active User (RSE DERTF) and DSO (A2A) interface (Sandroni et al. 2016, 3).

In Nguyen et al. (2016, 28–29), the interconnection of laboratories' microgrids has been conducted via cloud-based SCADA. A cloud-based SCADA system can be categorized either private, community or public according to the level of exposing data to the public. In private system, a SCADA application is set up on site, operated on an intranet and accessed only at the local level. Community system extends from the private system by allowing access to be granted for specific co-operators. The public system is completely run in the cloud via remote connection and requires access authentication. There are also three service delivery models for cloud computing, which should be considered. These service delivery models are IaaS, PaaS and SaaS. IaaS allows full control over applications and the infrastructure on the cloud for users. Further, PaaS permits users only to use but not control the applications. SaaS has the highest level of security and only enables users to use the application via a client interface, for instance, browser. Nevertheless, the operators are allowed full control over the infrastructure. (Nguyen et al. 2016, 28–29.)

The architecture of the presented system has been based on PaaS service delivery model and a hybrid cloud-based SCADA. Moreover, the applications have a direct connection with data analysis and the local control network has been conducted in the cloud. In addition, laboratories have had a local SCADA in their facilities for controlling and monitoring the laboratory components locally via industrial Ethernet. MTU and RTU have been used to connect the physical laboratory components to local SCADA, while standardized protocols, for instance, IEC 60870, IEC 61850 or DNP over TCP/IP, have been used for the communication. Each of these laboratory facilities has a microgrid platform, which has been connected to a common SCADA server on private cloud as presented in Figure 2.9. (Nguyen et al. 2016, 29.)

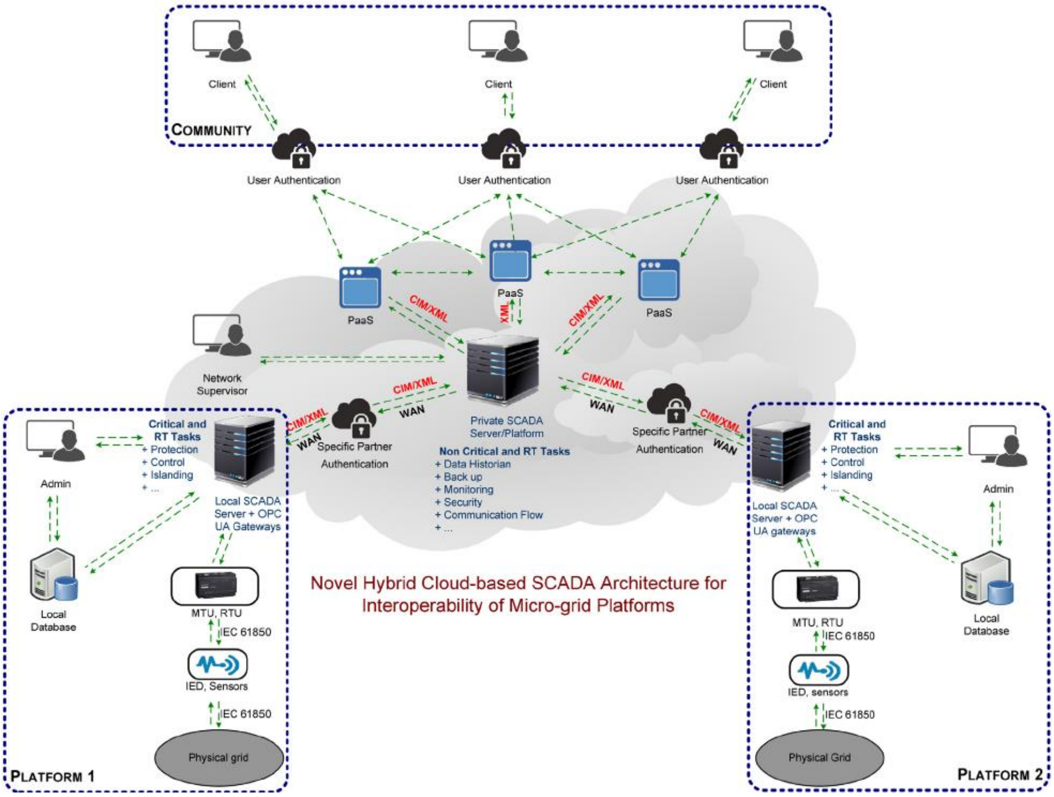

Figure 2.9. Interoperability between laboratory microgrid platforms via cloud-based private SCADA server (Nguyen et al. 2016, 30).

Data has been transferred from the physical devices of the laboratory microgrid platforms via local SCADA to the common private platform, which could be virtual PaaS/SaaS server or physical SCADA server. The connection between local SCADA and cloud-based SCADA could be based on WAN network and implemented as Webservice protocols and Ethernet with TCP/IP. The data transferred to cloud-based SCADA is specific non-critical information only, while all the critical functions have been enabled only on the local platform. The cloud-based SCADA server could be physically located at one of the platforms or other suitable site and monitored separate network supervisor. (Nguyen et al. 2016, 30–31.)

In DERri (2013a), the laboratory interconnection is implemented with Joint Test Facility for Smart Energy Networks with Distributed Energy Resources (JaNDER), which serves as a starting point for the ERIGrid JaNDER implementation. JaNDER is a concept of a common interface for laboratories, which enables data transfer between laboratory facilities. The interconnection is implemented by building JaNDER interface to each laboratory, which connects the laboratories to a joint SCADA via the Internet. Through the joint SCADA, it is possible to control and monitor the laboratories and conduct tests, in which equipment from multiple laboratories is used synchronously. There is a more detailed description of this concept provided in Chapter 5. (DERri 2013a; DERlab e.V. 2013, 26; DERri 2013b; CORDIS 2014, 9.)

2.6 Connection examples on communication stack

In the Sections 2.1–2.5, different communication protocols were discussed with the examples of remote connections and interconnections. The discussed protocols were from multiple different levels of ICT communication. Therefore, in this section, each of the previously discussed communication protocol is presented on a communication stack to clarify communication structures and relations between communication protocols.

Most of these communication protocols can be mapped to the (International Standards Organization / Open System Interconnection) ISO/OSI model. ISO/OSI model is a conceptual model for communication between two networked systems. ISO/OSI model was established to aid understanding of data communication and it divides the communication into seven layers, which together form the OSI stack. The seven layers are application, presentation, session, transport, network, data link and physical layer. Each layer has a different function and uses different units for data handling. (Alani 2014, 5–17; OSI 1994, 28–51.) The communication protocols discussed are presented in Figures 2.10–2.13 on the ISO/OSI model.

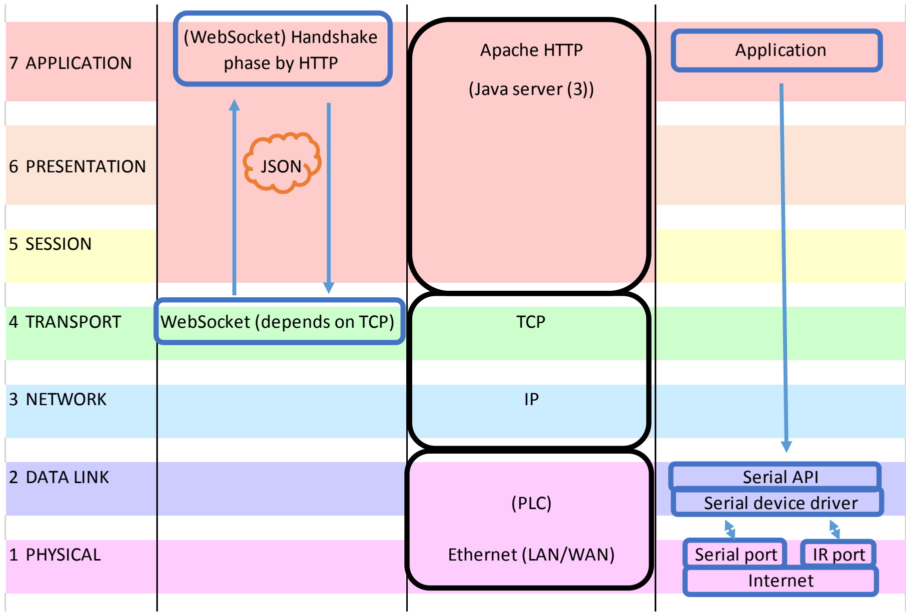

Figure 2.10. WebSocket, Apache HTTP with TCP/IP and Ethernet and serial communication to the Internet illustrated on ISO/OSI stack (Sáenz et al. 2016, 143–144; Hegedus et al. 2013, 257; Mostefaoui et al. 2017, 481–484; Besada-Portas et al. 2013, 156–158; Sendoya-Losada et al. 2016, 11503–11504, 11511; Mäkelä 2016, 31–33).

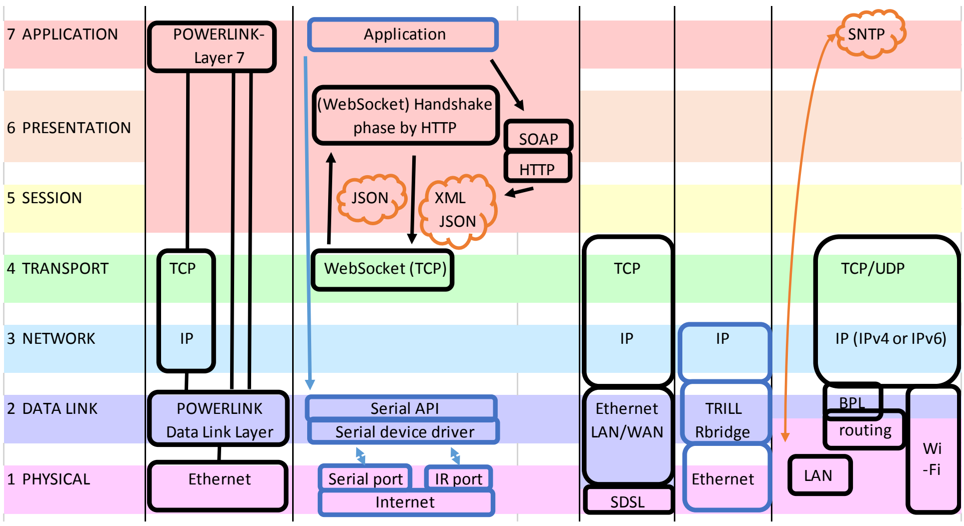

Figure 2.11. POWERLINK, iPower Flexibility Interface, SDSL and TRILL & RBridge illustrated on ISO/OSI stack (Wlas et al. 2011, 2070–2075; Orda et al. 2013, 314–316; Kolenc et al. 2017, 47–48, 50; Selga et al. 2013, 588–591; Della Giustina et al. 2011, 102–105; Della Giustina et al. 2013, 1–2).

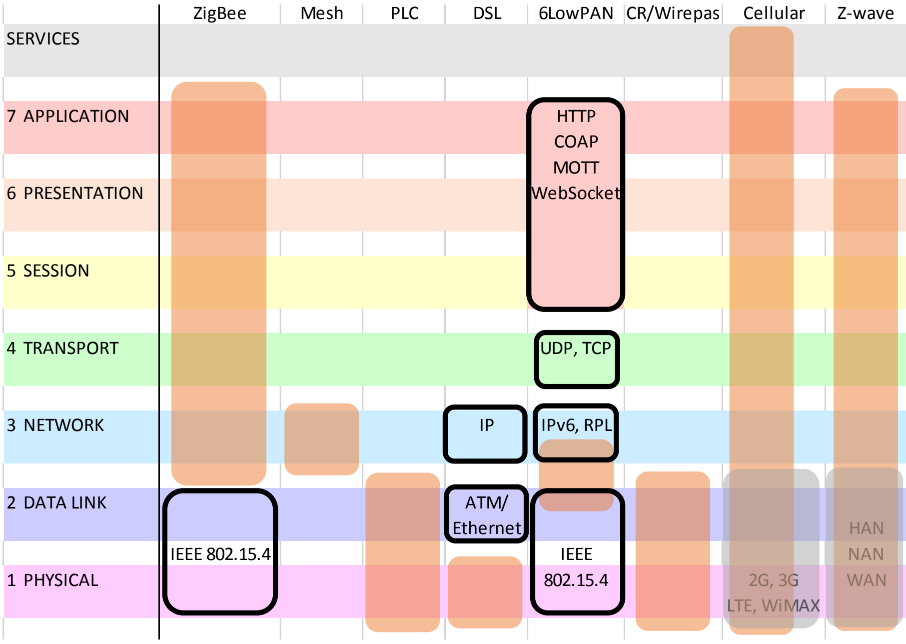

Figure 2.12. Smart meter ICT communications illustrated on ISO/OSI stack (Güngör et al. 2011, 530–534; Islam et al. 2016, 128; Khan et al. 2016, 860–863; Wirepas 2017, 3–5; Jacobsen et al. 2014, 131–133).

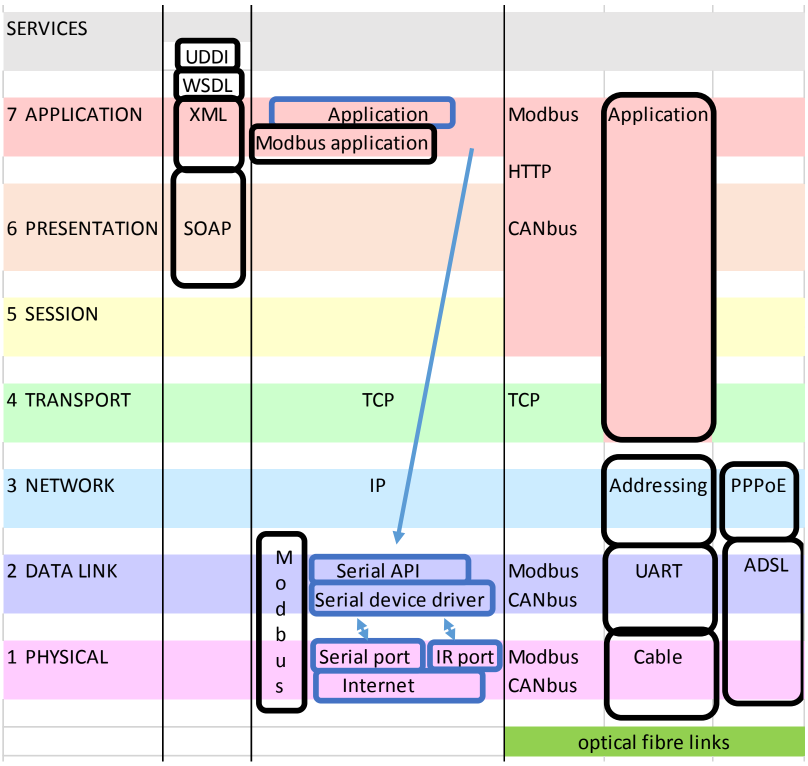

Figure 2.13. SOA concept and laboratory to microgrid connections illustrated on ISO/OSI stack (Leksawat et al. 2015, 1–4; Messinis et al. 2014, 31–34; Sandroni et al. 2016, 1–3).

In Figures 2.10 and 2.11, there are a couple of communication examples, which use TCP/IP. TCP/IP is a commonly used model for communication protocols and quite similar to ISO/OSI model. TCP/IP model was designed based on communication protocols already used, and therefore has become more popular than ISO/OSI model. (Alani 2014, 19–20.)

However, not all the communication protocols discussed in Sections 2.1–2.5 can be portrayed on ISO/OSI model as illustratively because of the advancement of the cloud computing technologies. Moreover, these communication protocols can be mapped on the Cloud Computing stack. Cloud Computing stack is a combination of all the different layers of communication needed to transfer data. These communication layers can be protocols or interconnected systems, which together form the path for data transfer. Unlike ISO/OSI stack, Cloud Computing stack is based on the type of services, since cloud computing utilizes different cloud delivery models based on the type of service. The Cloud Delivery Models are Infrastructure-as-a-Service, Platform-as-a-Service and Software-as-a-Service. (Erl, Mahmood & Puttini 2013, 25–40, 51–78.) Figure 2.14 presents the Cloud Delivery Models.

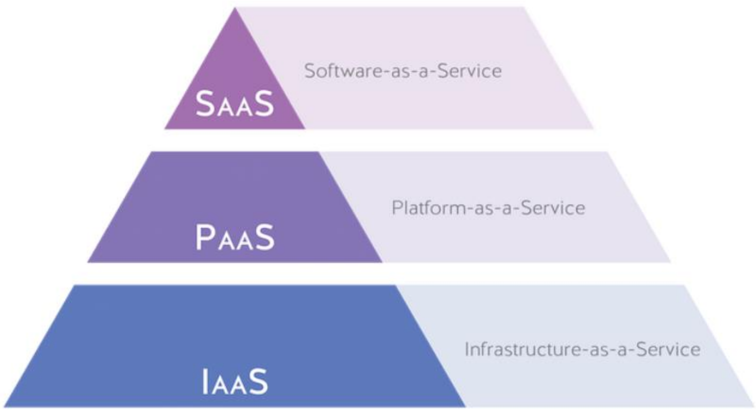

Figure 2.14. Cloud Delivery Models (Sandoval 2016).

In Nguyen et al. (2016, 28–31), the interconnection implementation has been based on cloud computing. JaNDER implementation from the research facilities is based on cloud computing and the implementation within the laboratory is based on TCP/IP and Ethernet connection in accordance with OSI stack (DERri 2013a; DERri 2010, 51–54).

2.7 Summary and lessons learned

In this section, a summary of the previous projects discussed will be provided. Additionally, lessons learned from the previous projects will be discussed with a comparison to JaNDER architecture and overall implementation.

Most of the remote connection and interconnection implementations were conducted for remote learning or other educational purposes, however, the importance of ICT implementations within smart grids is realised and slowly more research is conducted regarding smart grid remote connections and interconnections. Especially, DER remote connections are being researched. There are currently solutions available for implementing remote connection within both wired and wireless technologies. Wireless connections are often used due to their low implementation costs and high accessibility. While wired connections could be used in implementations with longer distances between the remote physical resources and rest of the smart grid, in which case wireless connections might be difficult to implement because of the interference. However, wireless connections are also used in long-distance implementation because of their lower investment costs compared to wired solutions.

In many of the implementations, using CIM or IEC 61850 based architecture or data model was considered to be the most suitable since both CIM and IEC 61850 are standardized and seems easily applicable for smart grid systems. In addition to CIM and IEC 61850, DLSM/COSEM (Device Language Message specification / COmpanion Specification for Energy Metering) is used as a common communication standard for smart meters (DLMS User Association 2007, 7; Kheaksong & Lee 2014, 391–393). DLSM/COSEM consists of object model viewing functionality of meter, an identification system for metering data, messaging method for converting the data to suitable format and a transporting method for transferring data (DLMS User Association 2007, 7). Although the number of interfaces and protocols suitable for implementation is quite large, some of the interfaces and protocols do not fill all the requirements needed for different applications in smart grids. Hence, combinations of two or multiple protocols were suggested also, as the combination might fill more requirements than a single protocol. Additionally, the large number of protocols and interfaces available currently could lead to difficulties in the implementation as not all the devices support all the protocols and interfaces. Moreover, the increasing amount of protocol conversions leads to very complex systems and because of this unification of the protocols and interfaces is needed.

Especially notable are smart grid requirements regarding security and reliability. For instance, locating DER or other smart grid units or components outside of the network was considered a great security issue, as the remote connection to outside of the network might be easier to access unauthorized and further gaining access to the entire smart grid. Implementing the interconnection via cloud service was considered to be one of the most suitable ways to avoid security issues. This has also been realised within the JaNDER implementation as the interconnection is planned to be conducted via a cloud-based real-time database, while previously a combination of VPN, gateway and routers was suggested.

Additionally, there are restrictions on the implementation interfaces and protocols caused by historical reasons. Since the power systems are currently filled with equipment, which is not necessarily very smart grid oriented, and replacing all that equipment is not possible because of monetary constraints, it was suggested that in the implementation should be also considered legacy equipment, for instance, legacy Ethernet devices, as the number of legacy devices in the power systems is high.

The smart meter communication was categorized to communication between appliances and smart meter and to communication between smart meters and DSO or to wired and wireless communication. The division was also conducted based on connection model between the units as a master-slave model or peer-to-peer model. For each type of communication, there are multiple technologies available, which each have their advantages and drawbacks, based on which the most suitable technology for each implementation could be chosen. Especially ICT system features, for instance, protocols, latency, communication network topology, bandwidth, information security and reliability, determine whether technology is suitable for the given implementation (Palensky et al. 2017, 17). Appendix A presents several protocols, their advantages, disadvantages and other features.

Especially for laboratory interconnections, implementations via a router or cloud-based SCADA were suggested by the state-of-art research. JaNDER has been previously implemented as an interconnection between routers via VPN Internet connection. However, the smart grid security requirements caused a difficult challenge in the implementation of the architecture with VPN and routers (Güngör et al. 2011, 534–535). As it has been realised that security requirements might be easier to overcome with cloud-based architecture, hence JaNDER has been developed to transfer data from local SCADA to a cloud-based database.

3 RELEVANT STANDARDS

This chapter provides an overall description of the standards relevant to this thesis. This chapter discusses IEC 61850 with additional details regarding points relevant for this thesis and CIM while also providing details regarding compatibility of IEC 61850 and CIM. Moreover, IEC 61850 and CIM are discussed as the implementation of the laboratory interconnection is conducted within the laboratory according to IEC 61850 and the structure of the laboratory grid is illustrated according to CIM. CIM was also needed in understanding the continuing discourses on the ERIGrid project.

3.1 IEC 61850

This section discusses IEC 61850 standard and provides brief overview and history of the standard. Additionally, the structures of the standard and the information model are provided alongside with discussion of benefits and security regarding the IEC 61850.

3.1.1 Overview and brief history

IEC 61850 is a substation communication standard based on Ethernet (Elgargouri et al. 2013, 1; Elgargouri, Virrankoski & Elmusrati 2015, 2461). Essentially IEC 61850 was established to enable interoperability between Intelligent Electronic Devices (IEDs) by different manufacturers from a regulation and standardization point of view (Elgargouri et al. 2013, 2). IEC 61850 differs from previously introduced communication standards in separating communication and application. Therefore, IEC 61850 is quite flexible in adapting to innovations of communication technology, as applications are maintained independently of any strict protocols and manufacturers can optimize their applications independent of the communication. (IEC 61850-1 2013, 16.)

IEC 61850 was established in 2003 by IEC Technical Committee 57 (Elgargouri et al. 2015, 2461). However, the foundation for IEC 61850 was laid in 1988, when the Utility Communication Architecture (UCA) was established. UCA determined the profile of suggested protocols for different layers of OSI model, data models and abstract services. UCA was used as a base for IEC 61850. (Mackiewicz 2006, 1.)

IEC 61850 was created for communication between devices within substations, but utilization of IEC 61850 has expanded since and currently the standard is utilized within substations and in smart grid communication. Utilization of IEC 61580 in smart grid solutions is becoming worldwide accepted (Mackiewicz 2006, 2; Kasztenny et al. 2006, 20).

3.1.2 Structure of IEC 61850 standard and additional standards

IEC 61850 consists of 14 parts (Elgargouri et al. 2013, 1). The overview of the IEC 61850 and linkages between each part of the standard are presented in Figure 3.1. Part 1 includes introduction and overview and Part 2 glossary. Parts 3–5 determine general and specific requirements for substation communication. Part 3 defines general requirements for IEC 61850 including quality requirements, environmental conditions and supplementary services. Part 4 describes engineering requirements, system lifecycle and quality assurance. Part 5 defines basic requirements and functions for communication, required logical nodes, logical communication links and performance requirements. (Mackiewicz 2006, 2; IEC 61850-1 2013, 20–21.)

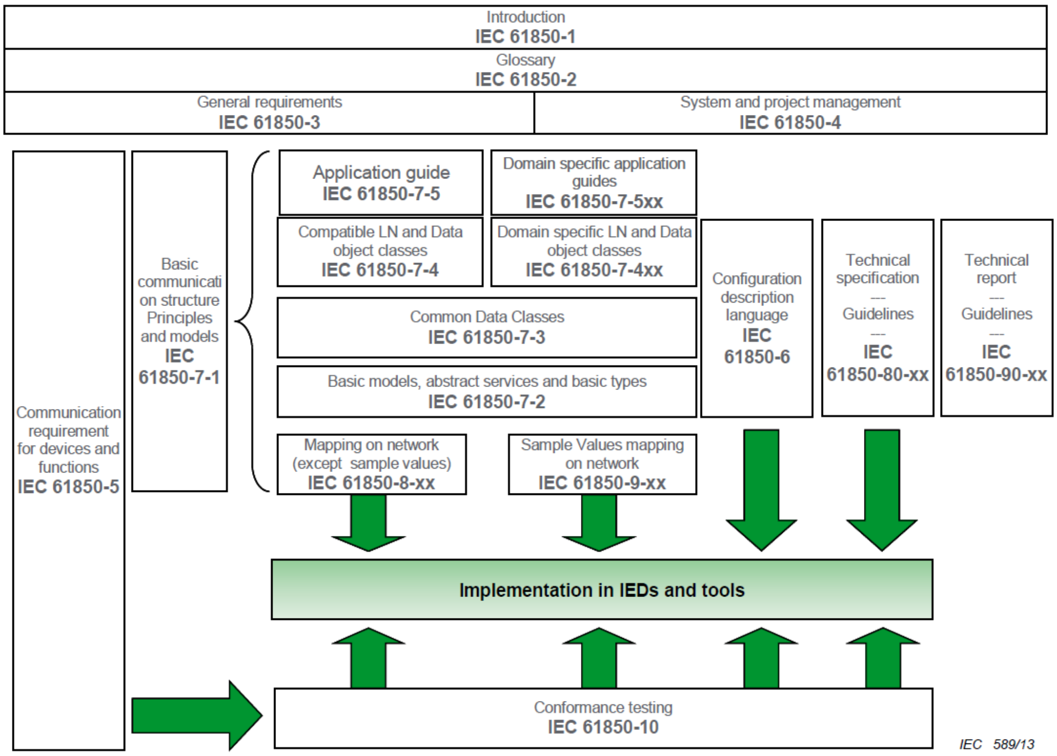

Figure 3.1. Overview of the IEC 61850 content (IEC 61850-1 2013, 22).

Part 6 defines Substation Configuration Language (SCL), which is established in XML (Extensible Markup Language) metalanguage. SCL is used to configure IEDs and describe the relations between substation and substation automation system. Part 7 consists of subparts 7-1 to 7-5. Subpart 7-1 defines principles and models, 7-2 abstract services the Abstract Communication Service Interface (ACSI) for information exchange and 7-3 Common Data Classes (CDC). Subpart 7-4 defines the abstraction of data objects for substation automation functions and 7-4xx additional domain-specific information models. Subpart 7-5 determinates usage of information models for substation automation applications and provides examples of data objects defined in 7-4. Subpart 7-5xx provides guides for other domain-specific applications, such as hydropower and DER domains. Part 8 consists of subparts 8-1 to 8-xx and specifies the implementation of mapping to Manufacturing Messaging Specification (MMS) and defines the implementation of all communication information except sampled values. Part 9 consists of subparts 9-1 and 9-2, which define the mapping of Sampled Measured

Values (SMV) to Ethernet data frame. Furthermore, all 9-x subparts are related to defining the mapping of sampled values. Part 10 consists of the description of conformance testing requirements and methods. In addition, parts 80-x and 90-x provide communication mapping Technical Specifications and further extensions Technical Reports respectively. Parts 80-x and 90-x also provide guidelines for utilizing IEC 61850 together with other standards, such as with IEC 60870-5-101 or IEC 60870-5-104 for communication between a control centre and substations or establish implementation, which would support IEC 61850. (Mackiewicz 2006, 2; IEC 61850-1 2013, 21–23.)

IEC 61850 is related to multiple other standards, and an interplay between these standards is possible. IEC 61850 has similar points in architecture with IEC 61499, which is standard on function blocks for industrial measurement and control systems, meaning that IEC 61850 logical nodes could be mapped to IEC 61499 function blocks. Similar architecture to IEC 61499 is utilized by IEC 61131, which is standard for communicating Programmable Logic Controller (PLC) inside substation automation. Therefore, IEC 61850 logical nodes could be implemented on function blocks of PLC system. IEC 60870 is utilized to transfer remote control and protection messages, and mapping IEC 61580 to IEC 60870-5 is under discussion, as it would increase the speed of wireless messaging since transferring GOOSE messages via wireless channels can be slow (as they were designed for non-wireless channels originally in IEC 61850). IEC 61850 also has additions such as IEC 61400-25, which can be used as an extension regarding wind power systems. (Elgargouri et al. 2013, 4.) IEC 61850-7-420 logical nodes could also be mapped to IEC 62351, as this would increase the reliability and security of the logical nodes since IEC 62351 is standard for data and communications security (Elgargouri et al. 2015, 2464).

3.1.3 Structure of the IEC 61850 information model

In IEC 61850, there are two major modelling levels in the information model, which are the division of the physical device into logical devices and of the logical device into logical nodes, data objects and attributes (IEC 61850-1 2013, 24).

Figure 3.2 presents the information model. The real device, which is connected to the network, is referred a Physical Device in IEC 61850. Each Physical Device can be identified by its IP address (Internet Protocol address). A single Physical Device could consist of one or multiple logical devices. (Mackiewicz 2006, 3; Baigent, Adamiak & Mackiewicz 2004, 4; IEC 61850-1 2013, 24.)

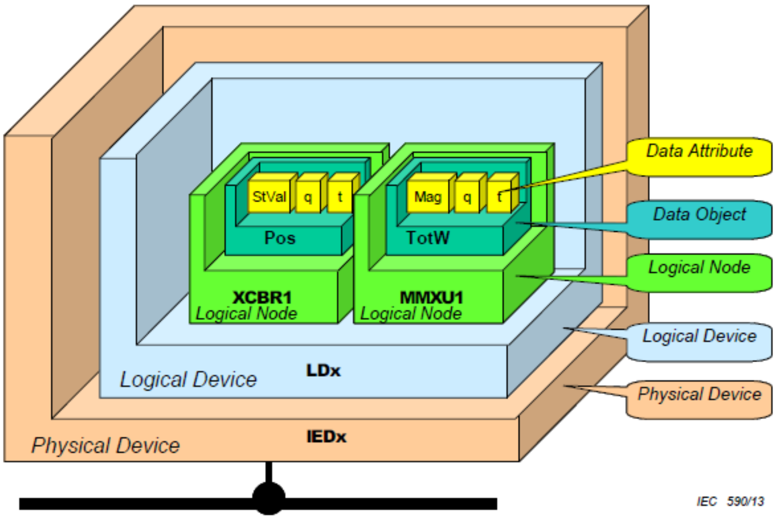

Figure 3.2. The Data Modelling in IEC 61850 (IEC 61850-1 2013, 24).

A Logical Device is a virtual presentation of Physical Device or parts of Physical Device, and more precisely each Logical Device portrays a group of certain functions in the Physical Device, for instance, automation functions or protection functions. However, Logical Device is not a virtual equivalent of a single actual real device. Although IEC 61850 does not determinate strict guidelines for the arrangement of Logical Devices in Physical Device, one Logical Device can be hosted by one IED only. The Logical Device acts as a communication access point. Moreover, Logical Device can transfer information about the Physical Device hosting them or external devices controlled by the Logical Device. Logical Device is a combination of one or multiple logical nodes. (Elgargouri et al. 2013, 2–3; IEC 61850-1 2013, 25.)

Logical Node is a named grouping of data, which is logically linked to a particular power system function. Logical Nodes are organized into groups based on their functions, for instance, all Logical Nodes related to automatic control have "A" as the first letter of the name. Table 3.1 presents the Logical Node groups. A Logical Node consists of data objects, which include data attributes (IEC 61850-1 2013, 25). Data Object is a data element with a unique name that illustrates the functionality and purpose of the data. The Data Objects are grouped into common data classes (CDC), in which Data Objects are organized based on the type and structure. There are CDCs such as status information, measured information, controllable status information and controllable analog setpoint information. Based on the CDC the Data Object belongs, it can have different attributes. The Data Attributes contain the actual information on status values, quality flags or time stamps. (Mackiewicz 2006, 3; Baigent et al. 2004, 4–5).

Table 3.1. Logical Node groups (IEC 61850-7-1 2011, 18).

Group indicator	Logical node groups
A	Automatic control
C	Supervisory Control
D	DER (Distributed Energy Resources)
F	Functional blocks
G	Generic function references
H	Hydropower
I	Interfacing and archiving
K	Mechanical and non-electrical primary equipment
L	System logical nodes
M	Metering and measurement
P	Protection functions
Q	Power quality events detection related
R	Protection related functions
S	Supervision and monitoring
T	Instrument transformer and sensors
W	Wind power
X	Switchgear
Y	Power transformer and related functions
Z	Further (power system) equipment

An instance of the information model structure based on Figure 3.2 is connected to the real network via the Physical Device, IED. The Physical Device consists of Logical Devices, such as the substation bay units or relays, which contain Logical Nodes. Logical Nodes could be for instance XCBR1 or MMXU1, within which the first letter of the name presents the logical node group. Based on Table 3.1, the “X” in XCBR1 illustrates that XCBR1 is a switchgear and the “M” in MMXU1 that the node is part of metering and measurement group. Furthermore, XCBR1 stands for circuit breaker and MMXU1 for 3-phase power measurement unit. Both of these example nodes’ names end with a number, this is to differentiate multiple of the same logical nodes in a logical device. Logical Node XCBR1 could contain Data Objects, Loc, OpCnt and Pos. Loc defines whether an operation is local or remote, OpCnt is used for operations count and Pos for the position of the switch. As Figure 3.2 shows, Data Object Pos includes Data Attributes, stVal as for status value, q for a quality flag and t for time stamp. Figure 3.3 presents a named structure of an example circuit breaker. (Mackiewicz 2006, 3; Baigent et al. 2004, 4–5; IEC 61850-1 2013, 25.)

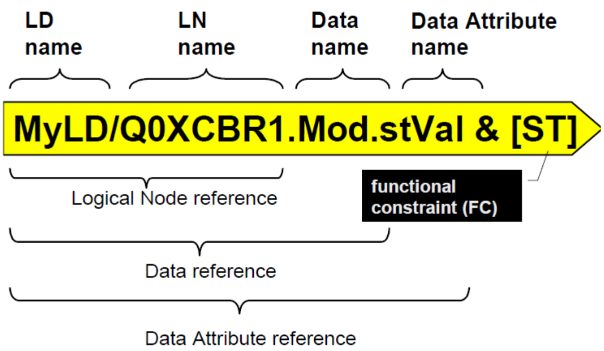

Figure 3.3. The structure of an object name in IEC 61850 (IEC 61850-7-1, 93).

In the structure of an object name in Figure 3.3, there is a Functional Constraint in addition to the structure parts presented above. Functional Constraint is used to define which services are applicable for each data class or data attribute. Based on the functional constraints of a data attribute, the possible services used to access information of the data attribute can be determinate. The functional constraint ST presented in Figure 3.3, for instance, represents status information, of which value can be read, substituted, reported and logged, but not written (IEC 61850-7-2 2010, 54–55). Table 3.2 presents all the possible functional constraints and services allowed. (IEC 61850-7-1 2011, 22, 54, 89, 93; Mackiewicz 2006, 3–4.)

Table 3.2. Functional constraint values (IEC 61850-7-2 2010, 54).

Functional Constraint	Semantic	Services allowed
ST	Status information	Reading, substituting, reporting, logging & non-writeable.
MX	Measurands (analog values)	Reading, substituting, reporting, logging & non-writeable.
SP	Setting (outside setting group)	Reading, writing, reporting & changes immediately effective.
SV	Substitution	Writing to substitute the value attribute, reading.
CF	Configuration	Reading, writing, reporting & changes immediately effective.
DC	Description	Reading, writing.
SG	Setting group	Used to maintain multiple grouped values of all instances of data attributes, non-writeable, current active value.
SE	Setting group editable	Edit buffer for value sets with SG constraint.
SR	Service response	Service tracking; reporting, logging, non-writeable.
OR	Operate received	The result of an Operate request at data object even with Operate execution blocked.
BL	Blocking	Used for blocking value updates.
EX	Extended definition (application namespace)	Used to define semantic definitions of LNs, data object class, data attributes, non-writeable.
XX	Representing all Data Attributes as a service parameter	All accessible data attributes, only used in functional constrained data (FCD).

All the data classes have set of functional constraints, which apply to the data attributes in these data classes. For instance, data class Single Point Status has functional constraints for status information, substitution, description and extended definition to constraint its data attributes status value, quality flag and time stamp. (Mackiewicz 2006, 3.)

3.1.4 Mapping and GOOSE

IEC 61850 mapping is conducted with Substation Configuration Language (SCL), which is established in eXtensible Markup Language (XML) and Unified Modeling Language (UML). XML is used to identify different elements within a document and thus enable utilization in the correct context. XML enables utilization of extensible markup, which allows usage of longer tags to describe all the components and measurements within substation in detail. UML is used in modelling processes, systems and devices and is standardized for specifying, visualizing, constructing and documenting both simple and complex systems. UML uses graphical notation and different kinds of diagrams to present designs of software and other systems. Benefits of UML include providing aid for team communication, for exploring potential designs and for validating architectural designs. Illustrating interfaces with standardized graphical designs and complex substation IEDs' models is enabled by UML. (Apostolov 2010, 1–2, 6.)

SCL is used to describe the relations between substation automation system and the substation. To configure the system, each device has to provide SCL file with a description of the configuration. (Mackiewicz 2006, 2.) SCL is described in IEC 61580-6. The main requirement for SCL is to guarantee interoperability between a station computer and IEDs, which could be all from different manufacturers. (Elgargouri et al. 2013, 3.) It is also possible to use SCL file in configuring some devices, which can process an import SCL file to configure automatically themselves, which can lead to time and cost savings in the configuration process. SCL defines the hierarchy of configuration files and therefore it is possible to use simple standardized XML files for configuration. SCL files include System Specification Description (SSD), IED Capability Description (ICD), Substation Configuration Description (SCD) and Configured IED Description (CID) files. SSD, ICD, SCD and CID have similar format, however, are used for different parts of the configuration. The main benefits of SCL files enable off-line configuration and system development and sharing of configuration files amongst suppliers and users. (Mackiewicz 2006, 3, 6.)

Several protocols can be used to map IEC 61850, but the standard includes mappings to MMS, GOOSE and SMV (Elgargouri et al. 2015, 2461). MMS is used for Station Bus transactions. MMS protocol is mapped on all the seven layers of OSI stack. MMS operates as a client-server protocol. MMS has also option for providing real-time communication mechanisms. SMV is used to transfer periodic time-critical data. For instance, synchronized streams of voltage and current sampling can be transferred with SMV. Both SMV and GOOSE are based on a publisher-subscriber model. Process Bus transactions are the most common application for GOOSE and SMV. (León, Montez, Stemmer & Vasques 2016, 2–3.)

GOOSE messages are used in the data exchange between IEDs, especially in the horizontal communication, in which same GOOSE message can be broadcast to one or multiple IEDs simultaneously. The version of GOOSE used by IEC 61580 allows transfer of datasets, while the previous version UCA GOOSE was designed to transfer binary only. IEC 61580 GOOSE can be used for sending analog data values directly, for sending data via virtual LAN and for setting the priority of the messages by a switch. Additionally, GOOSE can restrict logically the data flow to a specific broadcast domain. GOOSE messages are used for transferring sporadic time-critical data, especially protective relaying data. (Elgargouri et al. 2015, 2463; Kasztenny et al. 2006, 21–22.)

3.1.5 Benefits

The benefits of IEC 61850 are mostly resulting from the independent applications and communication, information model and GOOSE. The main benefit of IEC 61850 derives from requirements set for the standard and is the standard's ability to enable interoperability between IEDs by different manufacturers. The division to independent applications and communication creates an opportunity for vendors and utilities to update and optimise application functionalities. (IEC 61850-1 2013, 16, 25.) IEC 61850 communication and information models reduce maintenance costs and enable perfect performance inside SCADA because of the vertical and horizontal communications (Elgargouri et al. 2013, 1). The

IEC 61850 information model cuts down the amount of configuration since the model is organized in a consistent manner with self-description metadata, which automatizes creation of databases, a large part of the configuration. This automatic configuration creates time and cost savings. (Mackiewicz 2006, 2–3; Kasztenny et al. 2006, 21.) Benefits of GOOSE communication are less wiring, reduced costs and faster communication as GOOSE messages can be broadcast to multiple IEDs simultaneous (Elgargouri et al. 2013, 3). Overall, GOOSE messages bring many benefits such as reliability even in difficult communication conditions thanks to the non-deterministic character of Ethernet, enhanced monitoring, grid protection and reliability of control functions (Elgargouri et al. 2015, 2461).

IEC 61850 has also benefits regarding smart grids, which are mainly directly derived from the smart grid requirements set for the standard. These requirements include reliability, efficiency, flexibility and interoperability. The costs regarding installation, transducer and commissioning are decreased. Additionally efficiency of smart grids increases, since equipment migration, extension and integration costs are reduced with IEC 61850. Since the application and communication are not dependent, it is possible to add new capabilities, which improve the flexibility of the smart grid. IEC 61850 also makes insignificant or at least more feasible features, which have previously been available only limitedly. For instance, protection schemes and overall yielding of new applications has become less expensive because of incremental costs of accessing and sharing additional device data have reduced as based on IEC 61580 devices are connected to substation LAN. (Elgargouri et al. 2015, 2461; Elgargouri et al. 2013, 4.)

3.1.6 Security

IEC 61850 has quite a number of security issues and the standard itself should not be viewed as absolute secure (Elgargouri et al. 2013, 6). Although IEC 61850 has security issues, these issues are covered by other standard IEC 62351, which was formed by TC57 to solve security issues of IEC 61850 and all other TC57 series (Elgargouri et al. 2013, 6; IEC 61850-1 2013, 29). IEC 61850 security is covered by IEC 62351-6 (IEC 61850-1 2013, 29). A delay in communication is a major security concern of using IEC 61850 GOOSE messages. The delay is caused by the size of GOOSE messages. The threat of cyber-attacks and bugs has increased, since the mapping of IEC 61580 to Devices Profile for Web Services (DPWS) to increase the ease of interoperability of communication between devices inside a grid. IEC 61580 includes traditional handshake protocol to establish a communication path; however, this handshake is viewed to have a too long duration for real-time communication. Therefore, the enhanced handshake protocol is utilized in the real-time communication in smart grids. (Elgargouri et al. 2015, 2461–2463.)

There are solutions for increasing security thanks to the flexible structure of IEC 61850. These solutions are Intrusion Detection (ID) and enhanced handshake. ID essentially detects cyber-attacks; however, the detection protocol increases the duration of the delay. To reduce the duration of the delay enhanced handshake protocol could be used. In the enhanced handshake, the number of steps has been reduced to pair minimum, which could add to the security risks, since the enhanced handshake does not include all the authentication and authorization steps part of the traditional handshake. However, in the case of IEC 61850 enhanced handshake does not necessarily increase security risks, since according to the standard by default communication is only allowed when client and server have been authorized at MAC layer level. The reduced duration of delay increases the efficiency of the power system, while the cyber-attack risk decreases due to several milliseconds needed to execute a cyber-attack. (Elgargouri et al. 2015, 2463–2464.) In the standards, security measure provided for real-time communication is a digital signature as the duration of delay in most encryption and security methods is too long (IEC 61850-1 2013, 29).

3.2 Common Information Model

This section discusses Common Information Model and covers the structure, utilization and profiles of the model. Overview and a brief history are provided first in the section. Additionally, modelling languages and presentation of power systems based on Common Information Model are discussed.

3.2.1 Overview and brief history

CIM (Common Information Model) is an open standard for portraying power system components (EPRI 2015, 35). CIM was developed for EMS data exchange and transmission modelling, however, the scope of CIM has extended to cover objects and relations within distribution and generation also. Currently, CIM includes distribution modelling, electrical markets, meter management, asset management, customer billing, system planning, dynamic models, diagram layouts, geographical data exchange and work management. (Uslar et al. 2012, 23; McMorran, Lincoln, Taylor & Stewart 2011, 1.)

CIM consists of IEC 62325, IEC 61968 and IEC 61970-301 (Hargreaves, Pantea & Taylor 2014, 452). The aim of CIM is to define a uniform common model for defining all exchanged data between systems without duplicated definitions. Interoperability of data regardless of the data source is also a goal set for CIM. The motivation for a uniform standardized model is the savings as without uniform model the costs of integrating components would be significant. (Uslar et al. 2012, 6; Ivanov 2016, 22; McMorran et al. 2011, 1.) Since CIM includes all the exchanged data definitions, the model has grown extensively, because of this CIM could seem very complex at first. However, CIM is not designed to be used all at once, but as smaller sets of classes differing depending on the application. (McMorran et al. 2011, 1.)

CIM was first established by Electric Power Research Institute (EPRI) in the 1990s during Control Centre Application Programming Interface (CCAPI) project, in which the objective was to determine definitions for power system components to be used in EMS Application Programming Interface (API). To increase utilization, CIM was converted into several standards by IEC. (Uslar et al. 2012, 23–25; EPRI 2015, 35; Wollenberg et al. 2016, 128.)

3.2.2 *Structure*

CIM is a combination of several standards. The most significant standards are IEC 62325, IEC 61968 and IEC 61970, which each have a slightly different scope and applications. Figure 3.4 presents the overview of CIM parts. (Uslar et al. 2012, 75.)

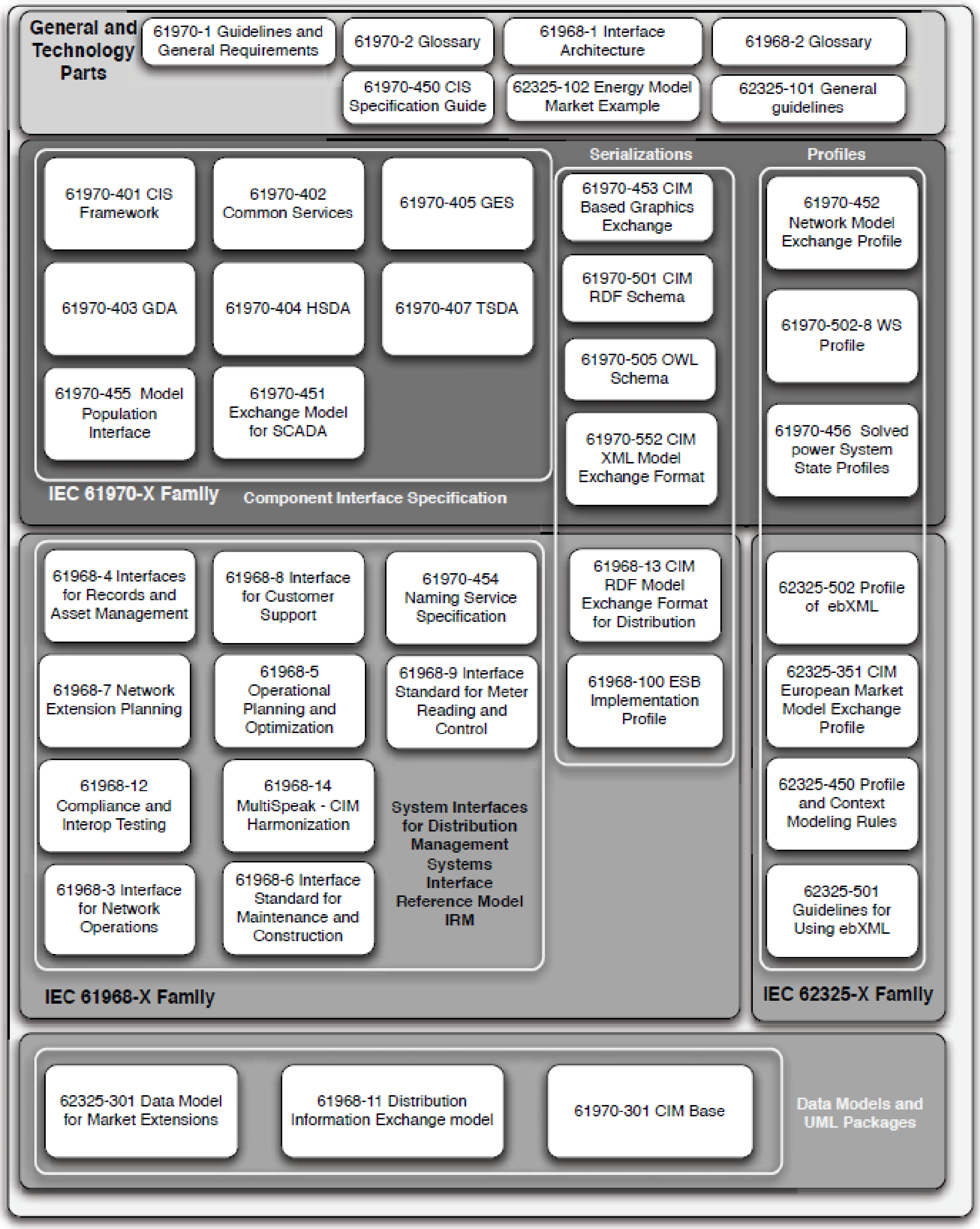

Figure 3.4. The overview of CIM parts (Uslar et al. 2012, 77).

EMS-API is included in IEC 61970, which is based on the EPRI CCAPI project. Multiple components are defined in IEC 61970. Developing a platform for independent data model by utilizing technology independent services is one of the main aims of IEC 61970. In the IEC 61970-301 subpart, an extensive data model is defined. This extensive data model includes descriptions of power system components and the relations between these components. (Uslar et al. 2012, 75–76; EPRI 2015, 25–26.)

IEC 61968 mainly consists of Interface Reference Model (IRM) for CIM. IRM comprises the description of use cases with specified details for coupling of two systems. IEC 61968 focuses on secondary objects used in grid operation, such as billing and network extensions, and especially IEC 61968-11 is an extension of IEC 61970-301 including definitions of additional data objects. IRM covers records and asset management, message exchange for network operations, operational planning and optimization, network extension planning, maintenance and construction, customer support and meter reading and control. (Uslar et al. 2012, 75–76; EPRI 2015, 25–26.)

IEC 62325 focuses on electricity market and the communication between market participants and operator and amongst operators. The standard divides markets into European style and US style markets. IEC 62325 describes data exchange for market operations, such as billing, clearing and settlement. Additionally, there are classes, which define agreements, invoices, prices, bids and notifications. (Uslar et al. 2012, 75–76; EPRI 2015, 25–26.)

3.2.3 Utilization

CIM is mainly used in system integration, custom system integration and serialization of topology data. CIM can be used as domain ontology for providing the vocabulary for business message exchange between system intra-utility. In this case, system integration is done by using predefined interfaces for distribution management system (DMS), IT and automation. There are pre-defined processes and payloads available as blueprints for standard processes. Custom system integration is possible with CIM since CIM has highly standardized semantics and syntax for exchanged business objects. Custom integration is also enabled by CIM's UML model, which allows model development and code generation. Additionally, applications, such as internal business object representation and storage could use CIM by using object-persistency stores. (Uslar et al. 2012, 23, 26.)

One of the other main uses for CIM is the topology data exchange. Graph of the power grid for topology exchange is established by serialization. CIM uses graph-like data structures and serialization rather than tree-like data structures, which makes examining and composing difference files for topology changes easier. As a result of the benefits of the graph like data structure and CIM overall, some countries have established utilization of CIM based topology exchange obligatory. CIM can be also used for market communication and exchanged data for European and US style markets. These applications regarding market communication can be enabled with extensions to the EMS. (Uslar et al. 2016, 26.)

The main benefits of CIM are derived from the flexibility as new objects can be added if needed. Furthermore, when these additional new objects are realized to be needed on a general level, they can be added to the standardized version of CIM thanks to the open nature of the standard model. (Hargreaves et al. 2014, 452.)

Because of the complex structure of CIM, organizations have been hesitant to adopt CIM. This is mainly because of ambiguity regarding standardization process, concepts and benefits of adopting CIM. (Ivanov 2016, 22.) However, CIM standards are accepted worldwide by the utilities and high level of standardization encourages more organizations to adopt CIM (Uslar et al. 2012, 3).

3.2.4 CIM profiles, classes and relations

Since CIM has a complex structure and not all the parts are needed in all the projects, CIM profiles have been created. Profiles consist of only part of the original CIM and only the essential classes, attributes and associations for the specific project are included. Profiles make utilizing CIM easier as a number of elements and scope is limited. (Uslar et al. 2012, 101.) The main benefits of utilizing profiles are consistency and uniformity the profiles create, while also enhancing and streamlining system integration activities (Österlund et al. 2016, 72).

Profiles have been built based on the data exchanges required for each interface. The most common profiles are Common Power System Model (CPSM), Common Distribution Power System Model, ENTSO-E and Electric Reliability Council of Texas (ERCOT). CPSM is used for the exchange of transmission system models in the US, while CDPSM is used for the exchange of distribution power system models and ENTSO-E for exchange transmission system models in Europe. ERCOT is an intra-corporate data model. The profiles are defined in IEC standards. Addition to the exchange of network models, there are profiles for message exchange between company and systems. (Uslar et al. 2012, 101–102, EPRI 2015, 98–99; McMorran et al. 2011, 2.)

A profile is built of essential classes, which portray some type of objects, such as transformer or customer. Moreover, each component in the power system is defined as a separate class. A class contains attributes and has relations with other classes. A certain class has always the same number and type of attributes and associations when instantiated, however, each instance of the class has its own internal values. Classes can be defined as subclasses of other classes, meaning that the subclass also contains all the attributes of the class, by which subclass it has been defined. This defining as subclass is called inheritance as the subclass, a child, inherits the attributes of the other class, a parent. A child class can also contain other attributes in addition to the inherited attributes. Classes can be either concrete or abstract. The class is abstract if it cannot be instantiated and serves just as a parent class for one or more child classes. On the other hand, concrete classes can be instantiated. (EPRI 2015, 41–42.)

Furthermore, classes can be also linked to one another by association, aggregation and composition. Association allows classes to use attributes of the associated classes and their child classes depending on the role of the association regarding each class. Associate classes can be created and deleted independently of the associations and each class has its own lifecycle. Aggregation is a special case of the association, in which a class acts as a container for other class. In the case of aggregation, there is ownership as each class can only belong to one container class, however, if the container class is deleted the classes aggregated to it are not deleted simultaneously. The composition is a special case of aggregation. In composition, the child classes do not have their own lifecycle; therefore, if their container class is deleted the child classes are deleted also. (EPRI 2015, 43–45.)

3.2.5 Languages and language schema

CIM is based on Unified Modeling Language (UML), so to understand CIM; one should have at least basic knowledge of UML. Addition to UML description languages eXtensible Markup Language (XML) and Resource Description Framework (RDF) are used in CIM. (Uslar et al. 2012, 47.)

UML is modelling and specification language. UML is used to model components in the software lifecycle, such as data structures, system interactions and use cases. UML based models can be used directly in electronic form by software tools and companies can just download them and combine with their own models and products. UML can be also applied to cybersecurity by definitions of users, companies, roles, consoles and function authorities. Authentication, authorization and separation are enabled by UML and specific architecture for cyber security can be defined. (EPRI 2015, 41; Wollenberg et al. 2016, 125; Skare, Falk, Rice & Winkel 2016, 95.)

XML is used to store human and machine-readable data in a structured, extensible format, which can be accessed via the Internet. XML is meta-language, which includes information on the formatting and context of the stored data. One of the main applications of CIM is message exchange, which is based on XML. (Uslar et al. 2012, 127; EPRI 2015, 47.)

RDF is an XML schema, which enables defining relations between XML nodes. As in basic XML, it is only possible to create parent or child connections between elements. With RDF, elements are assigned unique IDs under RDF namespace, which enables referencing other elements by their IDs. RDF does not include vocabulary for describing the relations. For the descriptions, RDF Vocabulary Description Language or RDF Schema (RDFS) is used. With RDFS, it is possible for the user to describe which classes or properties are connected. (EPRI 2015, 51–53.)

3.2.6 Connectivity nodes and terminals

In CIM, the connection between components in the power system is not defined by direct connections between components, but by utilizing connectivity nodes and terminals. In this node-breaker model, each component has one or more terminals, which each can be connected to one connectivity node. A connectivity node can have multiple terminals connected to it. Connectivity nodes are zero impedance points of interconnections and all terminals connected to the same connectivity node are therefore interconnected. Figures 3.5 and 3.6 illustrate the node-breaker model. (EPRI 2015, 63, 66; Österlund et al. 2016, 72.)

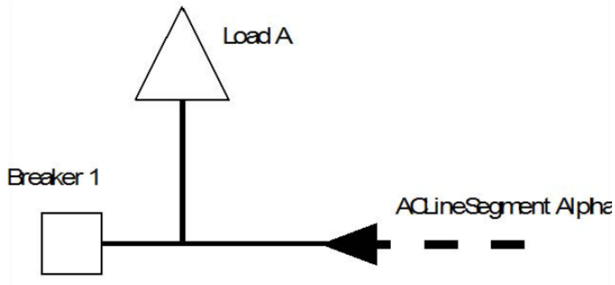

Figure 3.5. An example circuit (EPRI 2015, 63).

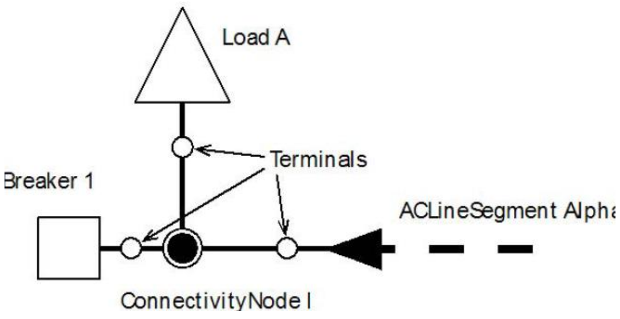

Figure 3.6. An example circuit with connectivity node and terminals (EPRI 2015, 66).

Figure 3.5 presents a simple example circuit including breaker, load and line. Figure 3.6 illustrates the same example circuit with connectivity node and terminals. The breaker, load and line are all connected to the connectivity node via terminal and each terminal is connected to a single connectivity node. Based on this example circuit, it might seem as utilizing only connectivity nodes without terminals would be sufficient. However, terminals are also used in determining measurements regarding points of connectivity, for instance, current flows and voltages. Figures 3.7 and 3.8 illustrate an example of a larger circuit as a single line diagram and CIM model.

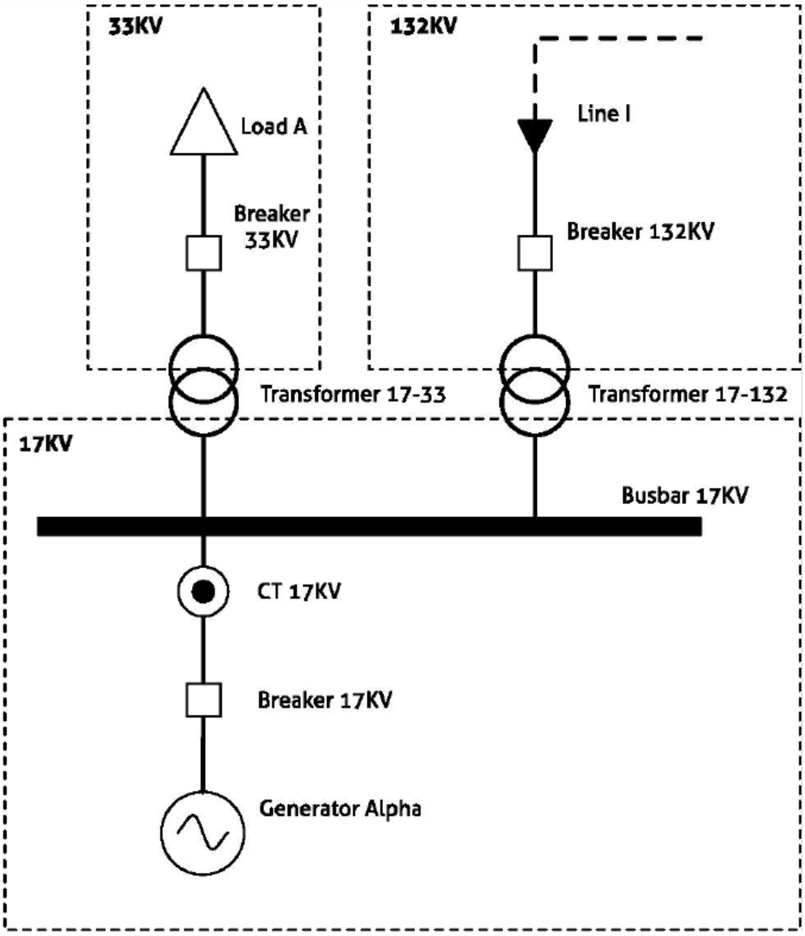

Figure 3.7. An example circuit presented as single line diagram (EPRI 2015, 67).

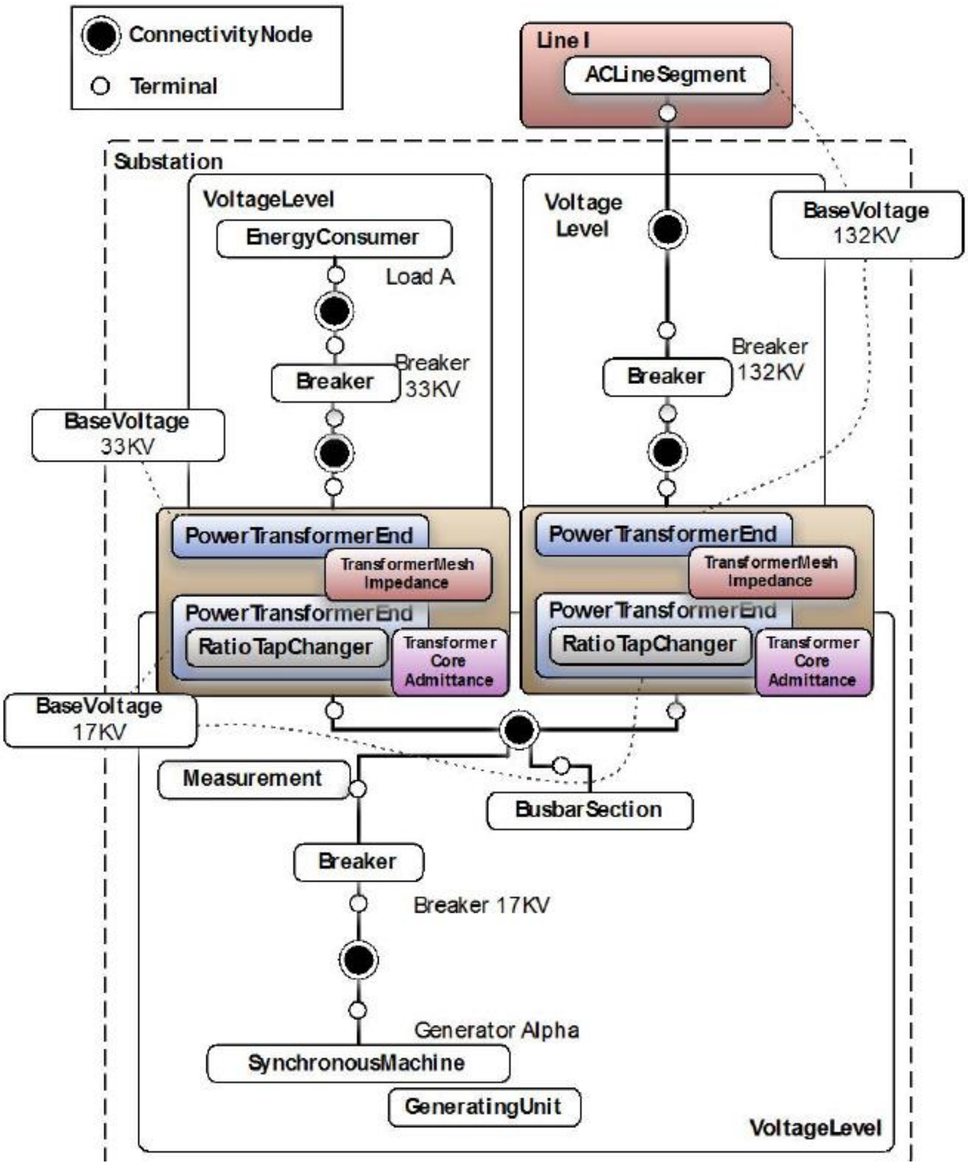

Figure 3.8. An example circuit presented as CIM model (EPRI 2015, 79).

Comparing illustrations of the example circuit in Figures 3.7 and 3.8, it can be noted that the transformer is mapped with more details in the CIM presentation. Moreover, the current transformer (CT) in Figure 3.7 is mapped in the CIM representation only as measurement instance, as CT does not affect the electrical characteristics of network or analysis results. (EPRI 2015, 80.)

3.3 IEC 61850 and CIM harmonization

IEC 61850 is used within substations for integration of IEDs and communication, self-describing equipment, advanced protection systems, substation automation and autonomous control. IEC 61850 contains detailed descriptions of substation equipment and controlling and monitoring the equipment. CIM is used within power control centres for model and data exchange and applications at control centre level. Static, dynamic and connectivity information of numerous equipment and substations is also included to CIM with detailed descriptions. (Pradeep et al. 2009, 1.)

Gallagher (2014, 245–246) and SGCG (2014, 184) have highlighted compatibility of IEC 61850 and CIM as one of the major issues hindering the development of future power systems. Compatibility is needed to maintain and increase the reliability and efficiency of power grids (Gallagher 2014, 245–246). Several solutions have been presented to harmonize IEC 61850 and CIM (Pradeep et al. 2009, 4–5; Santodomingo et al. 2013, 4350–4354; Mercurio, Di Giorgio & Cioci 2009, 1150–1153; Lee & Kim 2017, 201).

Pradeep et al. (2009, 4–5) integrated the standards by mapping objects. Each standard was separated into two files; IEC 61850 was divided into SCD and ICD file and object name structure of the data attributes file, and CIM was divided to a static file and a dynamic file. IEC 61850 and CIM were integrated by merging the static configuration files, which are SCD and ICD file for IEC 61850 and static file for CIM, and by merging dynamic time-stamped data object files, which are object name structure of the data attributes file for IEC 61850 and dynamic file for CIM. (Pradeep et al. 2009, 4–5.)

Santodomingo et al. (2013, 4350–4354) provided a solution for harmonizing IEC 61850 and CIM by conducting the signal mapping automatically with ontology matching techniques. For signal mapping, CIM, SCL and LN data models were needed. The harmonization process contained three steps. First, Web Ontology Language (OWL) ontologies were created for each data models, then the ontologies were aligned and lastly, the alignments were processed to map the signals.

(Santodomingo et al. 2013, 4350, 4354.) According to Santodomingo et al. (2013, 4354), this mapping system based on ontology was capable of identifying over half of SCL-CIM alignments.

Mercurio et al. (2009, 1150) provided a solution of IEC 61850 and CIM integration by creating extension based on CIM to Substation Automation System (SAS) data models from IEC 61850. For integrating the data models, a basic substation control extension of CIM provided by Electric Power Research Institute (EPRI) was used. The extension is especially targeted towards integrating IEC 61970, which includes CIM and IEC 61850-7-3 and IEC 61850-7-4. Additionally, new entities and attributes were added to describe all IEC 61850 information on IEC 61970 data models. (Mercurio et al. 2009, 1150.)

Lee et al. (2017, 201) harmonized IEC 61850 and CIM based on connectivity. In this solution, IEC 61850 and CIM were divided to model level and object level. Model level describes classes and object level demonstrates the instances of classes. The model level was utilized in the integration. The integration was conducted by first identifying data flow scenarios and entities within the standards involved in these scenarios and comparing the semantics of these entities. Then these entities were merged semantically based on CIM, the entities without a corresponding entity in the other standard were added and relationships between entities were established. Lastly, semantically similar entities were identified and merged with CIM based entities. This solution also included Query/View/Transformation (QVT) algorithms for converting IEC 61850 and CIM to the integrated model and vice versa. (Lee at al. 2017, 201.)

4 CASE LABORATORY

This chapter provides a general description of VTT's Multipower laboratory, the equipment available at the laboratory and for which type of projects Multipower laboratory could have potential. Additionally, the features of grid automation controller COM600 by ABB are discussed.

4.1 Multipower

The Multipower laboratory is a nationwide empirical research environment and provides opportunities for testing new technical DER products and solutions in a multifunctional environment (Mäki, Hashmi, Farin & Pykälä 2011, 1; AIT et al. 2014, 82; ERIGrid 2016, 2). Since the Multipower laboratory is a combination of multiple independent testing resources connected together, it covers areas of production, control and loading. There is local SCADA at the laboratory for measurements and control possibilities. (AIT et al. 2014, 82; ERIGrid 2016, 2.)

The laboratory network is 400 V low-voltage network. The laboratory network is connected to 20 kV public distribution network via 0.5 MVA and 1 MVA transformers. Additionally, an islanded operation of the laboratory is possible. The laboratory includes 1.6 MVA and 200 kVA diesel generators, 570 kVA brake dynamometer, 750 kVA converter, adjustable resistor loads up to 1700 kW and adjustable inductive loads up to 55 kVAr. Additionally, there is also a ready-made connection point for testing additional devices, 750 W PV panels and a grid emulator CINERGIA GE30, which can alter voltage and frequency of the feed-in for testing with different grid conditions and disturbances (Cinergia 2016, 1). With future updates, the laboratory could also offer possibilities for energy storage testing. Figure 4.1 presents an example of the Multipower laboratory configuration. Lastly, there is GPS system in the Multipower laboratory, which makes possible synchronization of remote units and related research by applying Simple Network Time Protocol (SNTP) (ERIGrid 2016, 2). (AIT et al. 2014, 82.)

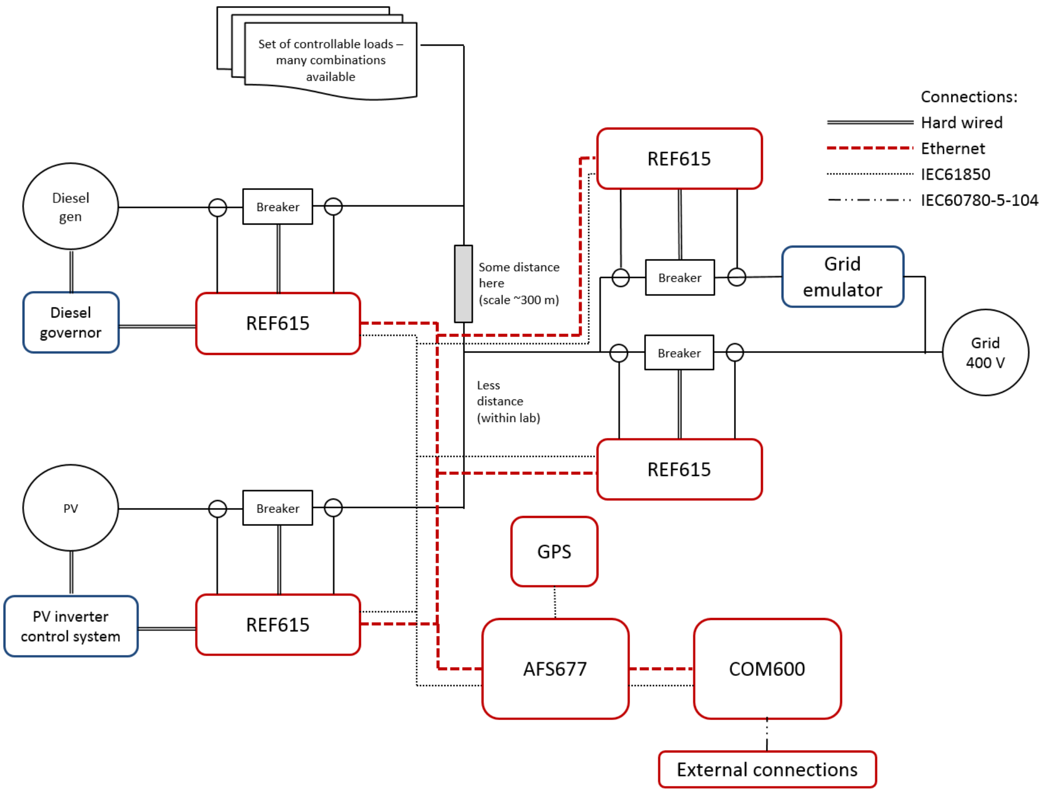

Figure 4.1. An example of the Multipower laboratory configuration (Mäki 2017) modified.

It is possible to conduct testing especially regarding IEC 61850 and communication at the Multipower laboratory. For instance, IEC 61850 protection or control solutions and GOOSE messaging can be tested with the substation automation system, which consists of ABB COM600 substation computer and four ABB REF615 feeder protection IEDs. Regarding communication, the Multipower laboratory is part of a 5G pilot area as VTT is participating in 5G development. (ERIGrid 2016, 2.) Ideally, DER generation units, ICT applications for DER systems, distribution automation and devices based on IEC 61850 and malfunctions and faults could be tested at the Multipower laboratory (ERIGrid 2016, 3).

4.2 COM600

ABB's COM600 is a substation automation and management unit. It is possible to build a virtual substation with COM600 and run both simulations and physical power system with it. COM600 has gateway functions for mapping signals between protection and control IEDs. COM600 also contains HMI for transferring data to users and providing visualization of the substation as a single line diagram. It is also possible to access HMI remotely via web HMI. (ABB 2017, 15.)

COM600 is compatible with several communication protocols and supports interoperability between different protocols. Hence, gathering data by connecting IEDs with various protocols is possible. COM600 includes web technology for displaying data and transferring data to Network Control Centre (NCC) or Distributed Control System (DCS). COM600 uses IEC 61850 SCL and IEC 61580 data modelling for communications modelling. (ABB 2017, 15.)

In the Multipower laboratory, the COM600 is connected to four REF615 feeder protection IEDs, which are further connected to a PV unit, a grid emulator, the distribution grid and a movable connection point for additional devices respectively. Via REF615 devices, it is possible to cultivate data from the generation and load units and transfer the data to COM600.

COM600 has Gateway functionality for enabling data transfer and connections between COM600 and OPC servers or clients. Figure 4.2 displays an abstract diagram of the gateway concept. COM600 uses Station Automation Builder 600 (SAB600) for connecting OPC Clients to NCCs and OPC Servers to protection relays and network equipment. (ABB 2017, 16.)

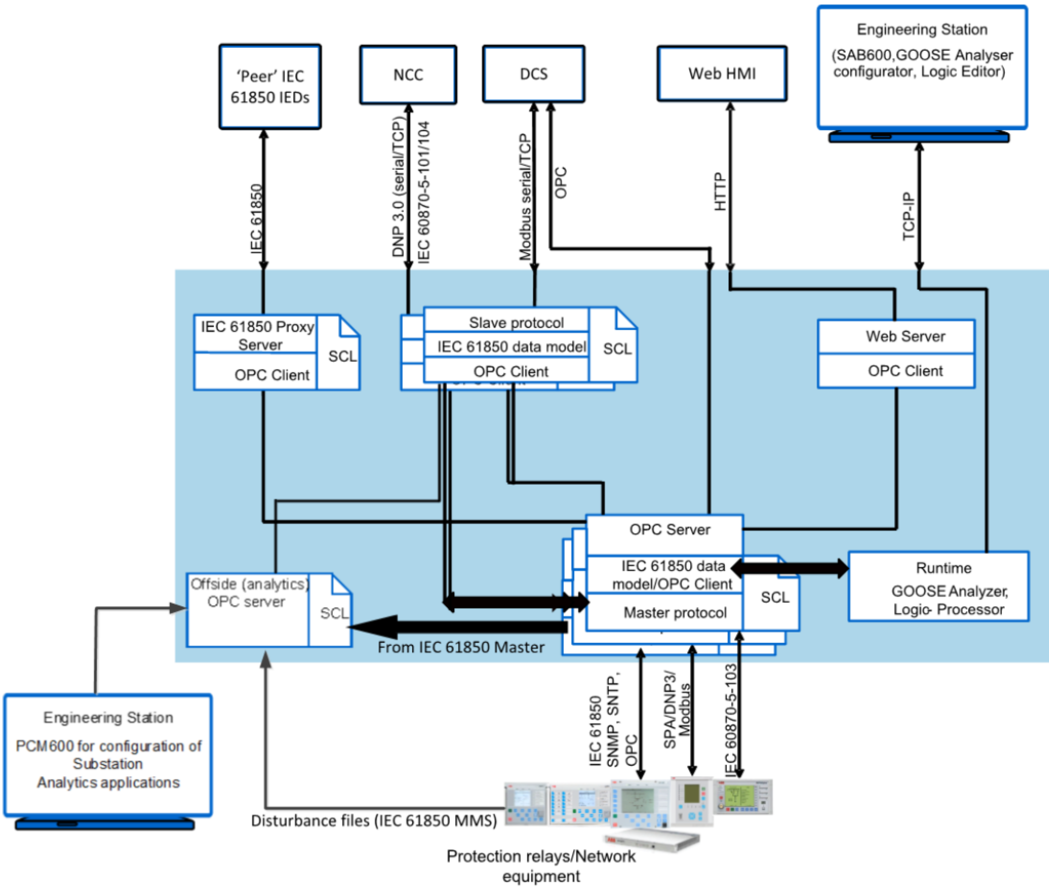

Figure 4.2. Conceptual presentation of the COM600 gateway, where NCC is Network Control Centre and SAB600 Station Automation Builder 600 (ABB 2017, 16).

Additionally, COM600 has a PLC system based on IEC 61131-3 standard. The PLC allows programming and executing automation tasks within COM600. The PLC programming is conducted with CodeSys SP runtime, thus information from OPC Server is transferred to other OPC servers, clients and WebHMI via CodeSys SP runtime. (ABB 2017, 20–21.)

IEC61850 standard is used as a foundation for protocol conversation and signal mapping in COM600. Each client and server is a separate program and interacts with other components only via OPC interfaces. Hence, server and clients can be developed and used independently. Since OPC does not specify the semantics of the data, the same information could be available in different formats depending on devices and communication protocols used. To avoid this type of situation, the data modelling is implemented according to a structure defined in IEC 61580 as presented in Figure 4.3. (ABB 2017, 22.)

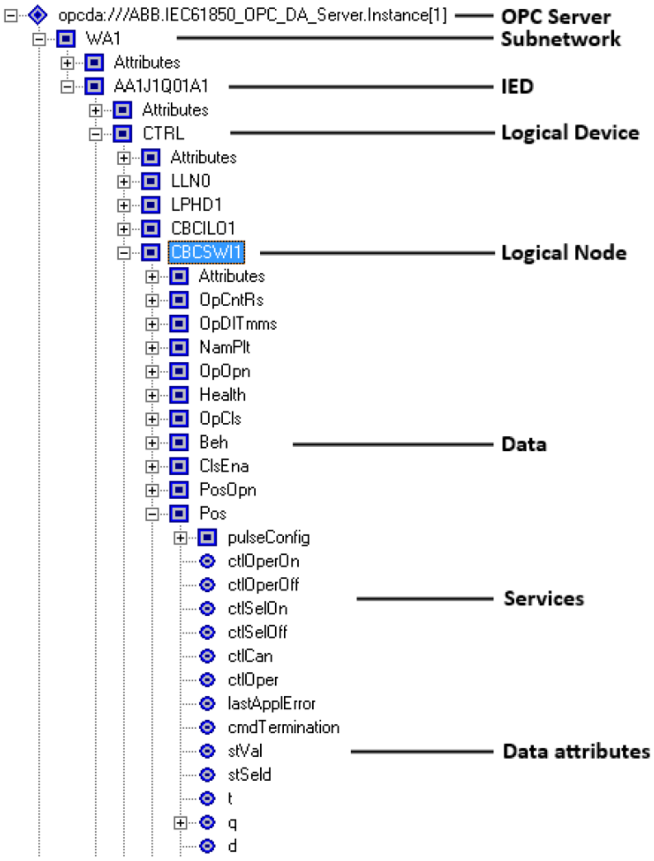

Figure 4.3. COM600 communication structure according to IEC 61580 data modelling (ABB 2017, 24).

Additionally, COM600 includes Data Historian and Station/Remote switch function. Data historian is a Real-Time DataBase (RTDB), which stores real-time and history values. Data historian can be used for storing, analysing and presenting process data. Station/Remote switch function is used by OPC clients and servers to determine control position of IEDs' logic devices. Station/Remote switch can be physical one or simulated one. When the switch is in station position controls from OPC, clients are rejected. (ABB 2017, 21–22, 24.)

ABB COM600 uses IEC 61850 in the substation structure. The substation structure is a diagram displaying the major component and connections to the components within a substation. Substation structure can be built manually, based on Connectivity packages or based on SCL files. At least gateway, substation voltage level and bay objects should be added to the substation structure. Substation structure can be built only after the communication structure has been built. Busbars, bays and connections between components can be added on Single Line Diagram (SLD) Editor. Additionally, display boxes and indicator for measurements and operational modes can be configured on SLD Editor. (ABB 2017, 46, 49.) Figure 4.4 presents an example of a substation structure on SLD Editor.

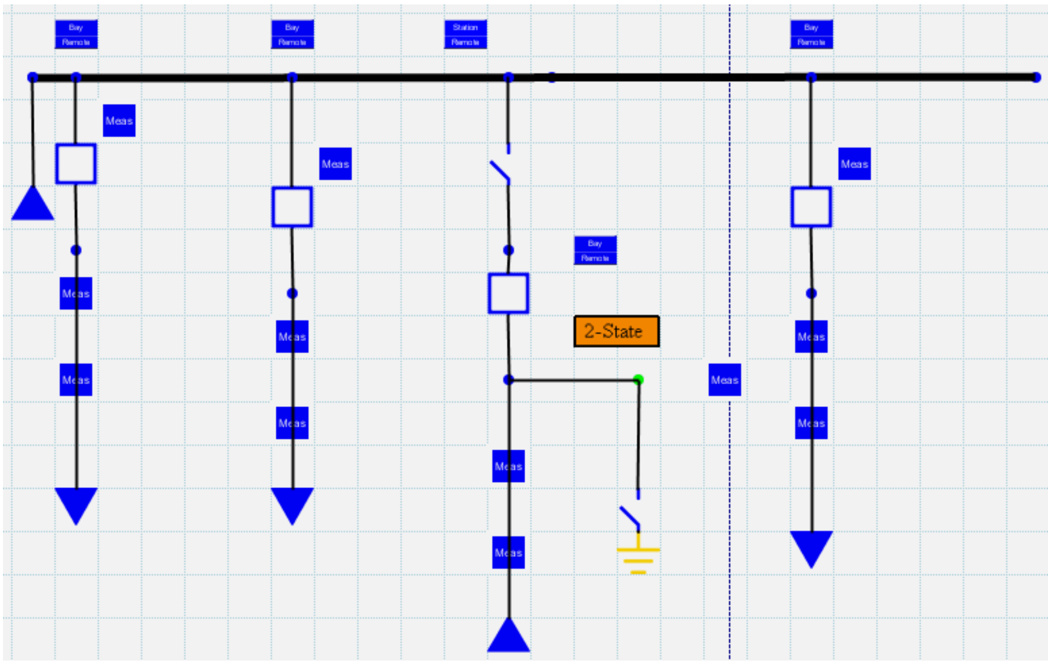

Figure 4.4. An example of COM600 substation structure on SLD Editor. Blue boxes are measurement displays and blue dots represent terminals.

As Figure 4.4 presents, components in the substation structure are connected based on IEC 61850 using terminals. The substation structure is displayed and can be monitored via HMI single line diagram. (ABB 2017, 49.)

5 JANDER

This chapter discusses JaNDER protocol by explaining the historical background and architecture of the protocol. Additionally, topics of communication, security, benefits and development of JaNDER are covered. The current implementation of JaNDER is covered in Chapter 6.

5.1 Overview

JaNDER is a concept for implementing interconnection between laboratories. Moreover, as laboratories are interconnected it is possible to perform joint tests, simulations, remote monitoring and measuring. JaNDER is foremost established as a common interface for controlling DER and measuring system data in laboratories (DERri 2013a). The first version of JaNDER was developed during DERri (Distributed Energy Resources Research Infrastructure) project as one of the JRAs. One of the objectives of the JRA and the project was to create a demonstration of interconnection of multiple laboratories' testing equipment via a single communication protocol (DERri 2013a). JaNDER has been conducted to improve utilization of test facilities and to enable testing on with equipment of multiple laboratories, which would allow the creation of more complex test cases, for which single research facilities would not have the necessary equipment and infrastructure. Additionally, JaNDER has also been conducted to provide transnational access to research groups without the needed testing equipment to allow these external researchers to conduct tests at the research facilities. Figure 5.1 illustrates JaNDER concept. (DERlab e.V. 2013, 26.)

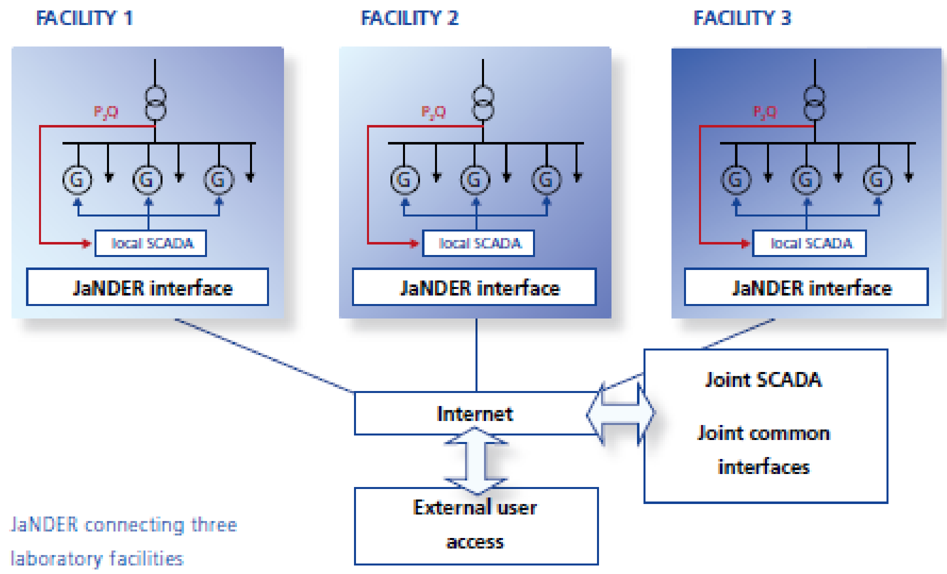

Figure 5.1. JaNDER concept with connections to three test facilities (DERlab e.V. 2013, 26).

JaNDER gateway has linked the communication interface to the laboratory infrastructure. JaNDER gateway has been based on IEC 61850 and a practical solution, which was mostly established on available software packages. As Figure 5.1 presents, there has been a joint SCADA in addition to the local SCADAs of each facility. The joint SCADA has accessed the facilities via a common interface, which has allowed access independent of the local SCADA. JaNDER has also included time synchronization feature used between physical laboratory and remote test equipment. The time synchronization has been accomplished with real-time repositories installed in every JaNDER gateway in all the laboratories. Practically, the gateway has been built between MMS protocol and an ad hoc protocol, eXtensible Modelling and Control environment (XMC) by RSE. Furthermore, the gateway has acted as an interface between the testing facility and the local LAN. The gateway could be thought to operate as SCADA in providing an immediate view of the topology and of the principal electric values for the operator (DERri 2013b). Each laboratory has had their own JaNDER gateway, which they could configure in the most suitable way for their laboratory. (DERlab e.V. 2013, 26; CORDIS 2014, 9.)

5.2 Architecture

JaNDER architecture and the implementation has been based on VPN via Internet connection operating between the research facilities, while each research facility has had a gateway for connecting to the VPN. The gateway has acted similarly to SCADA as it would provide a view of the topology and principal electric values of the facility to the operator in real-time. Via the gateway, it has been possible for remote users to receive values from the research facility and also modify them. (DERri 2012, 5.)

As Figure 5.2 presents, the JaNDER gateway has been connected to a JaNDER remote user via the Internet. JaNDER remote user has had IEC 61850 client software for interacting with the remote operator and VPN software to forward the protocol through a secure private channel over the Internet. On both the side of the JaNDER remote user and the JaNDER gateway there have been routers, which have been connected by VPN over the Internet. Local Graphic User Interface (GUI) console has been used to monitor the communication locally and included a browser for rendering and animating Scalable Vector Graphics (SVG) schemas. (DERri 2012, 8–9.)

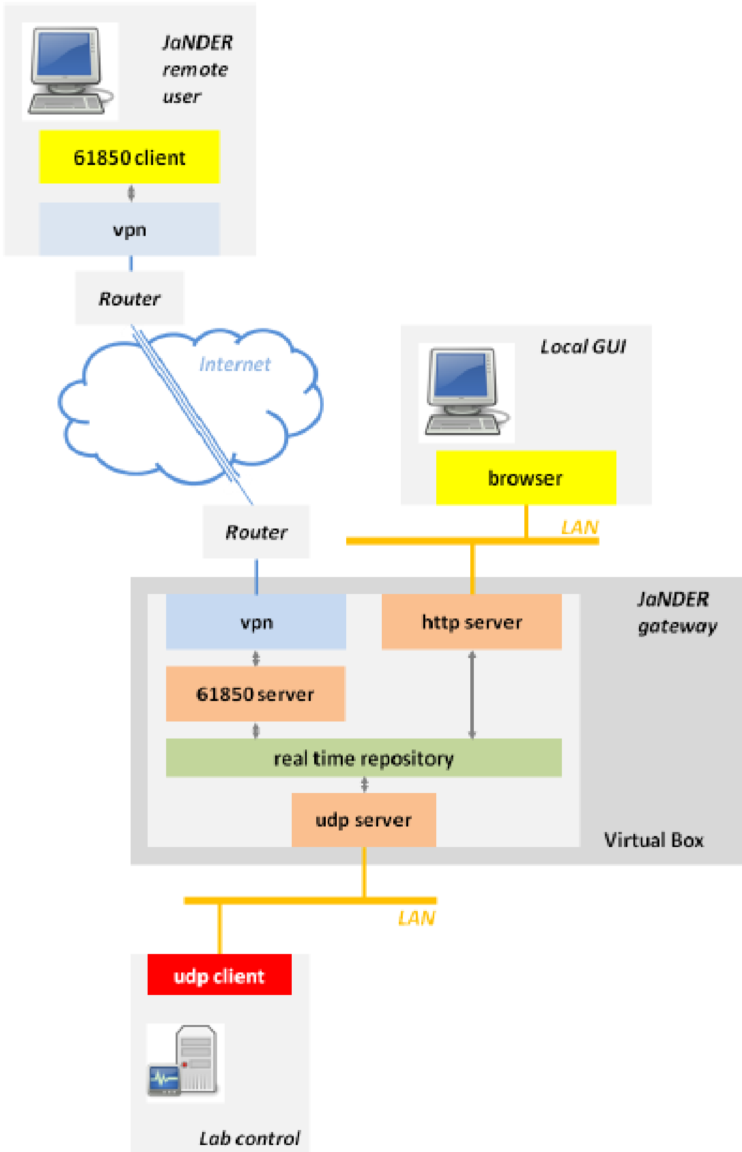

Figure 5.2. JaNDER architecture and possible implementation of the interface (DERri 2012, 9).

The actual JaNDER gateway has included multiple parts. The real-time repository has had a shared memory area for hosting the data received from the remote operator and from the laboratory in a format suitable for IEC 61850 protocol. The 61850 server has supported reading and writing to the repository by remote clients. VPN software has provided remote users secure private channel over the Internet and HTTP server has supported local GUI clients. User Datagram Protocol (UDP) server has provided the real-time repository with measures from the laboratory and forwarded commands from remote users via XMC protocol. The research facilities have had a control system, which had a UDP client for sending measurements and receiving commands from remote users. (DERri 2012, 9–10.)

5.3 Communication and security

In communication, ICT security has been addressed especially regarding confidentiality, integrity, availability and accounting objectives. Figure 5.3 presents the ICT security in the communication layout. (DERri 2010, 51–52.)

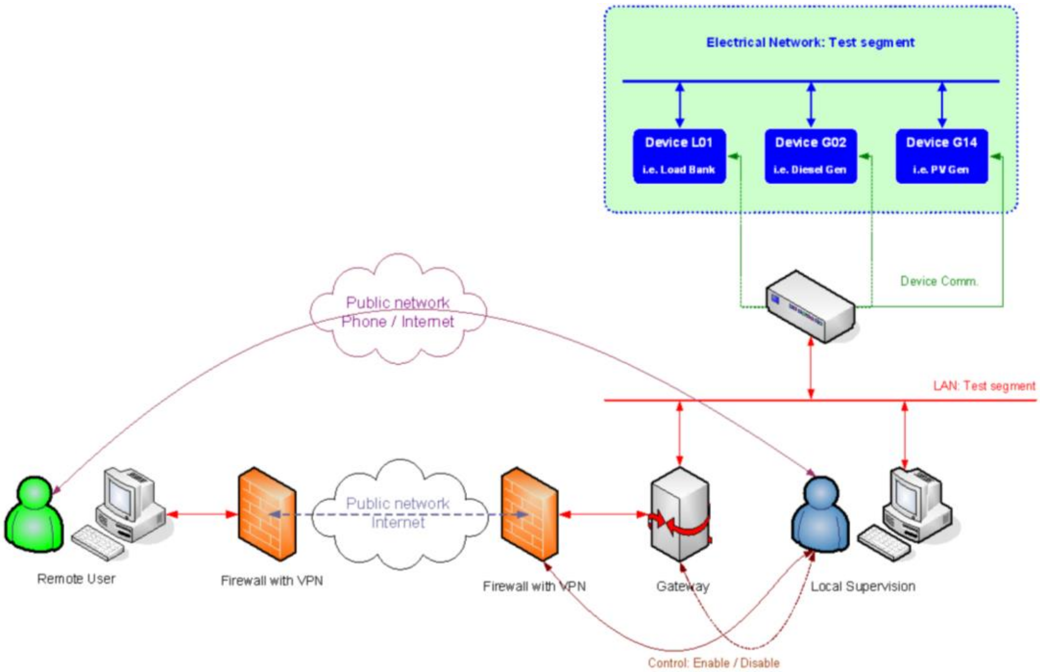

Figure 5.3. JaNDER communication layout (DERri 2010, 52).

The security aspects have been considered at several points of the communication layout, for instance at LAN Test segment, Gateway and VPN. LAN Test segment safety requirements have allowed only isolated equipment connect to a specific LAN segment to avoid unintentional access and cross commanding. These safety requirements could be implemented by using separate network element and swapping network terminals. Moreover, lockouts, tagging and staff warnings should be used if the safety requirements have not been met. The gateway has an assigned server for interfacing remote connections, which has reduced the tasks of the remote user, as they would not need to implement all the private communication protocols every time new devices have been added or become available. In the gateway, there has been also filters to deny execution of any offending commands. Additionally, a same common standard should be used for defining the data model, communication protocol and interfaces, and ultimately the local operator should have full control of the gateway to allow and deny commands from remote access users. (DERri 2010, 53–54.)

VPN has been used to connect research facility and the remote user. VPN was chosen as it uses the Internet, which is easily accessible, as a communication channel and has several different solutions and implementations to meet all the safety and other requirements of the implementation. Additionally, using the easily accessible Internet as a communication channel has allowed low-cost implementation in comparison to private communication networks. VPN could be implemented by software, hardware or open source solutions, out of which open source solutions VPN has had point-to-point security, which has allowed only authorized remote users to access the cryptographic keys for accessing the local network. Moreover, the cryptographic keys should have expired, when the access has been no longer necessary, to enhance the safety of the communication even further. Similarly as the gateway, a local supervisory for VPN have ultimately enabled and disabled remote access. (DERri 2010, 53.)

5.4 Benefits and development

There have been multiple benefits of implementing JaNDER. The remote interconnection could be accessed via the Internet. The number of equipment available for testing has increased due to the access to the equipment in multiple laboratories and to the testing facilities offering possibilities for a wider range of testing with a variety of equipment. Interconnection of laboratories has increased the utilization of the laboratories and also efficiency while providing better access to other laboratories and laboratory collaboration. Furthermore, the efficiency of implementing research has raised as each laboratory has followed the same standardization and procedures. (CORDIS 2014, 9.)

There has also been a disadvantage of utilizing JaNDER, the implementation process of the gateway and VPN connection has been quite difficult because of security requirements of the research facilities (CORDIS 2014, 9). Moreover, a major challenge during the DERri project was especially the integration of remote software, remote simulators and hardware (VTT et al. 2017, 13). To diminish the disadvantages of JaNDER found during DERri project, JaNDER has been improved and developed within the ERIGrid project. The ERIGrid project is especially addressing the difficulties in the installations caused by firewall changes required, lack of test cases utilizing several research facilities, the insufficient participation by partners and the disadvantages of gathering raw real-time data without context (Gehrke 2017, 12). The JaNDER architecture has been improved, since the previous architecture with VPN, gateway and routers was too challenging to implement at some research facilities, as it required modifying the firewalls and overall IT security at the laboratories.

In the ERIGrid project, JaNDER architecture is based on real-time databases (RTDB). RTDB is database using real-time computing to manage constantly changing data (Buchmann 2005, 104). The implementation consists of a local RTDB and central cloud-based RTDB and replication between these databases via HTTPS connection, which allows implementation of JaNDER without alterations to the IT security and firewalls at the laboratories. Since DERri project, three slightly different JaNDER interfaces, JaNDER level 0, level 1 and level 2, have been developed (VTT et al. 2017, 12). JaNDER level 0 is based on replicating the data between local RTDB and a cloud-based RTDB. JaNDER level 1 is based on IEC 61850 protocol and data model for transferring individual data, for instance, measurements and control signals. JaNDER level 2 is based on CIM and the entire CIM model of the research infrastructures is used in the data transfer (VTT et al. 2017, 12). These improvements have been conducted to increase the number of laboratories with JaNDER implementation and simplify the implementation of JaNDER to new resources to increase the probability of succeeding in the implementation (AIT et al. 2014, 55).

6 IMPLEMENTATION

This chapter discusses the implementation process of real-time laboratory interconnection. Information regarding installations, programming and testing of the interconnection will be provided.

In ERIGrid, there are three implementation levels of JaNDER, which Figure 6.1 presents. JaNDER level 0 is the first implementation level and in this stage, the remote laboratory interconnection is done by transferring the data to a local real-time database from where it is replicated to a central real-time database in a cloud. In the JaNDER level 1, the interconnection is implemented with IEC 61850 protocol and information model based interface connected to the central real-time database. While in JaNDER level 2 the similar interface is created based on CIM protocol connected to the central real-time database. Only, implementation of JaNDER level 0 is within the scope of this thesis. Figure 6.2 presents the architecture of JaNDER level 0 implementation.

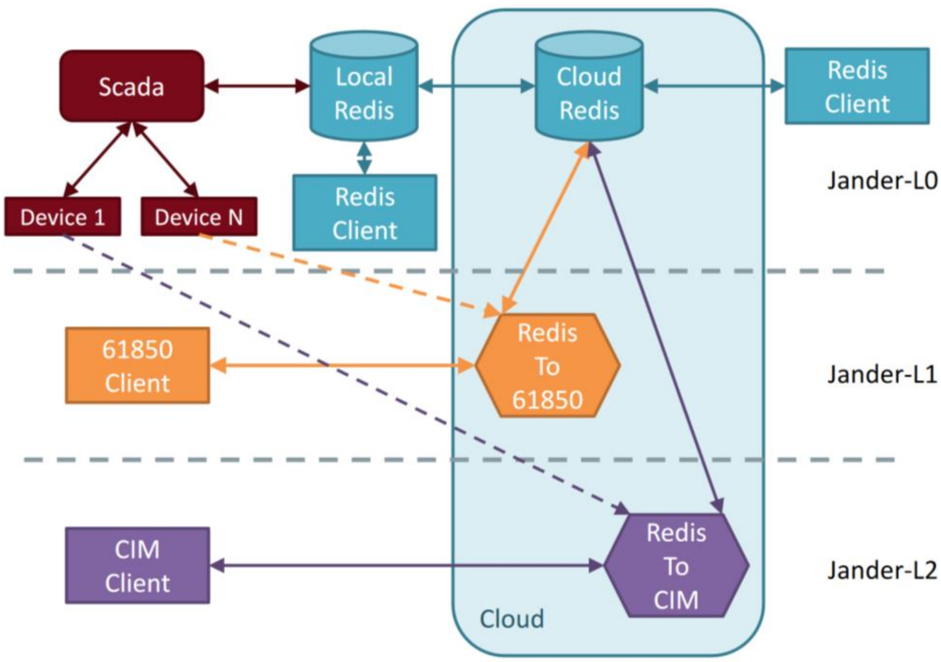

Figure 6.1. JaNDER architecture levels used in ERIGrid for implementation of the remote laboratory interconnection for integrated and joint testing (Gehrke 2017, 13).

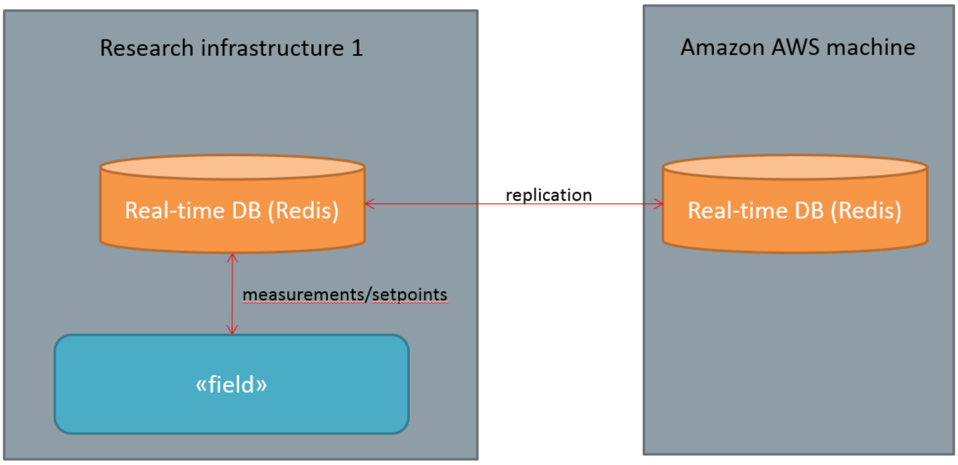

Figure 6.2. JaNDER level 0 architecture for implementation of the remote interfacing (Sandroni, Verga & Pala 2017, 18).

As Figure 6.2 illustrates, there is a local real-time database installed in the laboratory through which the data are transferred. The real-time database chosen for the implementation is Redis because of its easy to use functions and flexibility. The measurement and control data are transferred between field, which could be for instance SCADA, and the local Redis real-time database and further replicated to the central real-time database in a cloud, where the data from all the laboratories participating to ERIGrid is replicated.

6.1 Installations and configurations

In this section modification of the physical network at the laboratory and the installation and replication connection of Redis RTDB will be discussed. Additionally, information on configuring COM600 to connect to the relays and to the external connection will be provided.

The main challenge in this thesis was the establishment of the connection between COM600 and Redis. In the laboratory, the physical network routings between COM600, AFS677 router and the laboratory laptop were inadequate, which was realised quite late only. The network routing was correctly installed and configured only for LAN 1, which is used for the remote configuration and management of the COM600 from the laboratory laptop (ABB 2017, 40–41).

However, LAN 2 for the remote connection of communication and data transfer had not been routed at all (ABB 2017, 40–41).

The additional routing was conducted by establishing second Ethernet connection between COM600 and the laboratory laptop for the LAN 2 connection. Moreover, the Ethernet cable for the LAN 1 connection was removed from the Ethernet port of the laboratory laptop and connected to the laptop via a TP-Link Ethernet network adapter. The LAN 2 connection was connected to the LAN 2 port of the COM600 and directly to the Ethernet port of the laboratory laptop as Figure 6.3 portrays.

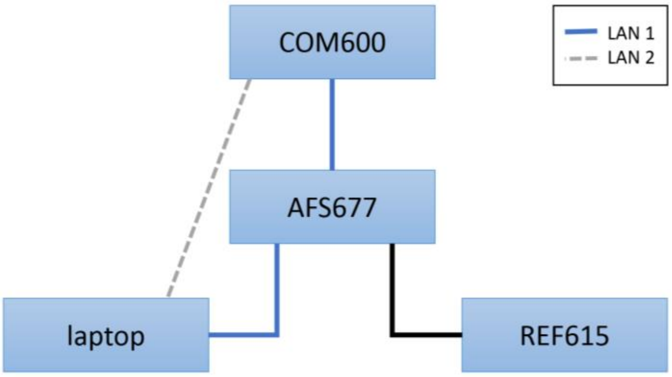

Figure 6.3. Physical network routing between COM600, AFS677, REF615s and laboratory laptop at the Multipower laboratory.

It was not possible to establish a connection between COM600 and Redis during the time of the inadequate routing. The missing connection from LAN 2 caused several errors and difficulties in the operating the program for connecting COM600 and Redis. For instance, the program reported the Start of Frame (SOF) indicator missing.

6.1.1 Substation automation and management unit

ABB COM600 needed to be configured to transfer data from electrical process to the NCC. The configuration was conducted in three parts. First, the connection between electrical process and COM600 was configured, an IEC 61850 information model was built to illustrate the electrical process, which then could be transferred to the OPC Server of COM600. The second step was to configure the gateway between COM600 and NCC, which was conducted by configuring an IEC 60870-5-104 Slave to transfer information between OPC Client and NCC. For this implementation, IEC 60870-5-104 Slave was chosen over IEC 60870-5-101, since the protocol is better suited for long-distance connections. Lastly, the OPC Client and OPC Server needed to be connected inside COM600 by configuring Cross-Reference function. It is important to note that the configuration of the COM600 Gateway could be conducted only in this order since the OPC Server has to be configured before adding OPC Client to the Gateway. Figure 6.4 presents the connections from electrical process to COM600 to NCC. (ABB 2017, 51–55.)

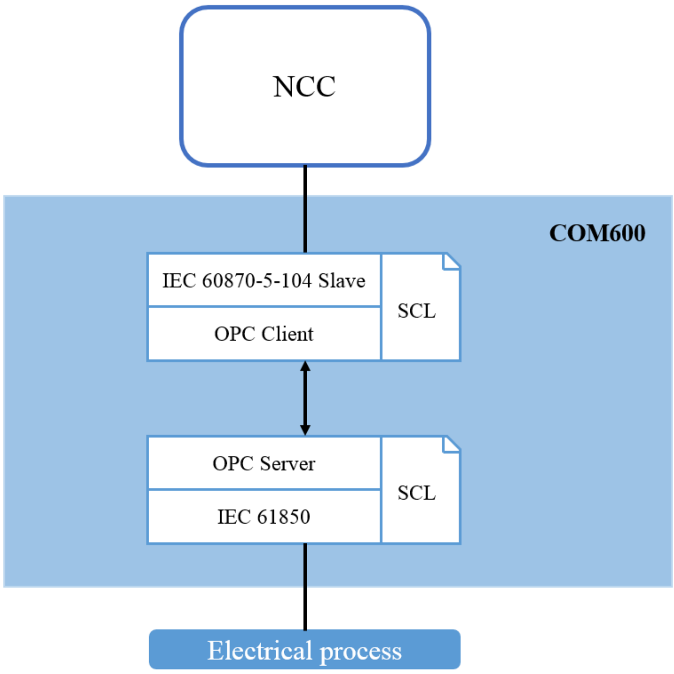

Figure 6.4. COM600 Gateway used in the implementation of the real-time laboratory interconnection, where NCC is network control centre and SCL substation configuration language (ABB 2017, 16) modified.

To build the process communication, the communication structure had to be assembled by adding objects to the object tree. The objects were to be added following the same structure presented in the IEC 61850 information model. The objects were added in the following order: Gateway object, OPC Server object, Channel object, IED object, Logical device objects, Logical node objects and Data objects. The communication structure below IED could also be built automatically via connectivity package or SCL description file. After adding the objects presented in Figure 4.1 to the communication structure, the structure looked as portrayed in Figure 6.5. (ABB 2017, 22, 51–55.)

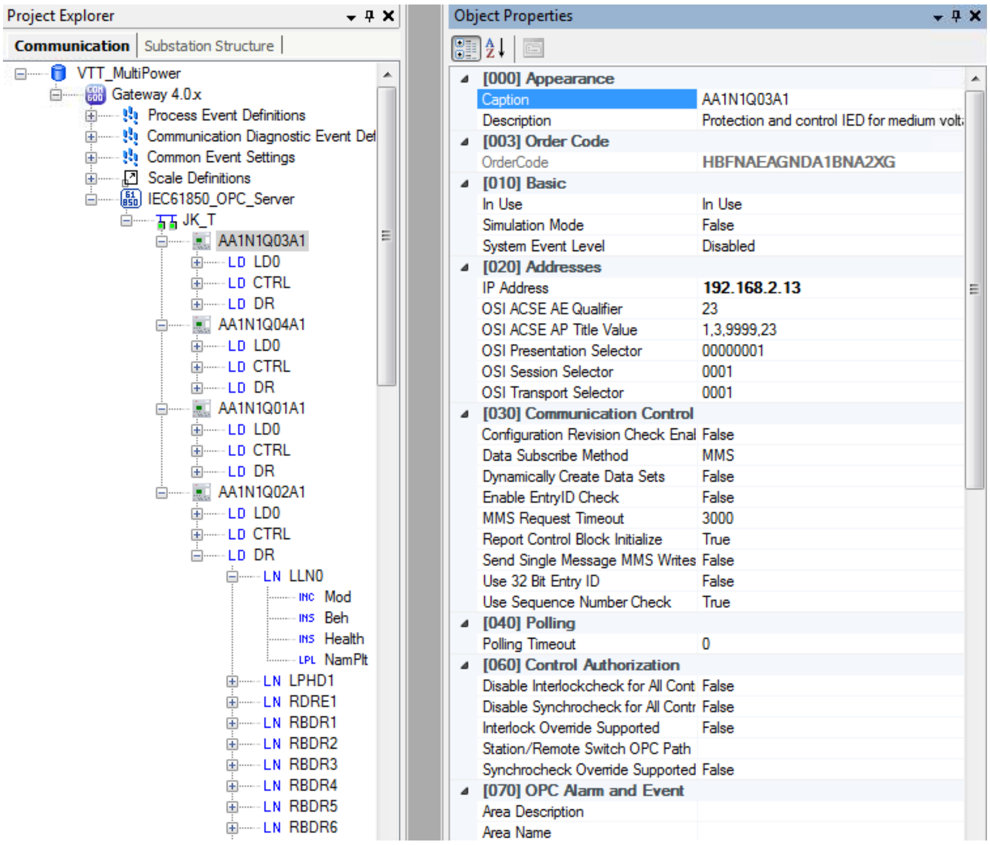

Figure 6.5. The communication structure on the left side, where IEC 61850 Subnetwork is the Channel object, AA1N1Q01A1, AA1N1Q02A1 and AA1N1Q04A1 are IED objects, LD0, CTRL and DR are Logical device objects, LLN0, LPHD1, RDRE1 and so forth are data objects and Mod, Beh, Health and NamPlt are data attributes. On the right side, object properties table of an IED for object configuration.

During the building of communication structure, most of the objects had to be configured via separate object properties, an example of which Figure 6.5 provides for IED object properties. When the communication structure was configured, the configuration had to be activated in Management dialog of the COM600 by selecting Management on the Gateway object and choosing Update & reload configuration. It was also possible to check if the communication between COM600 and the IEDs was working by selecting online diagnostics on the bay and viewing the connection status in the status information section. (ABB 2017, 51–55.)

After building the communication structure, the substation structure had to be finalized. Substation structure could be finalized by creating connections between busbars and slightly modifying SLD layout if communication structure had been implemented by utilizing Connectivity packages or SCL description files, otherwise, substation structure had to be created manually. It was important to overview the substation structure created while combining communication structure since there could be additional connectivity nodes or other objects in the substation structure, which should be removed. While modifying the substation structure additional objects caused a slight challenge, as at first, the substation configuration was not valid because of a number of validation errors. According to the validation errors on the Output window, there were excess nodes added during the implementation via Connectivity packages. The issues were solved by manually removing the excess nodes from the substation structure. As a part of the finalizing, busbars needed to be created and connected to bays to arrange the voltage level layout. The communication structure and the substation structure could be linked together with data connection function, which enables data object information to transfer between the two structures. Figure 6.6 presents the final substation structure and SLD layout. (ABB 2017, 51–55.)

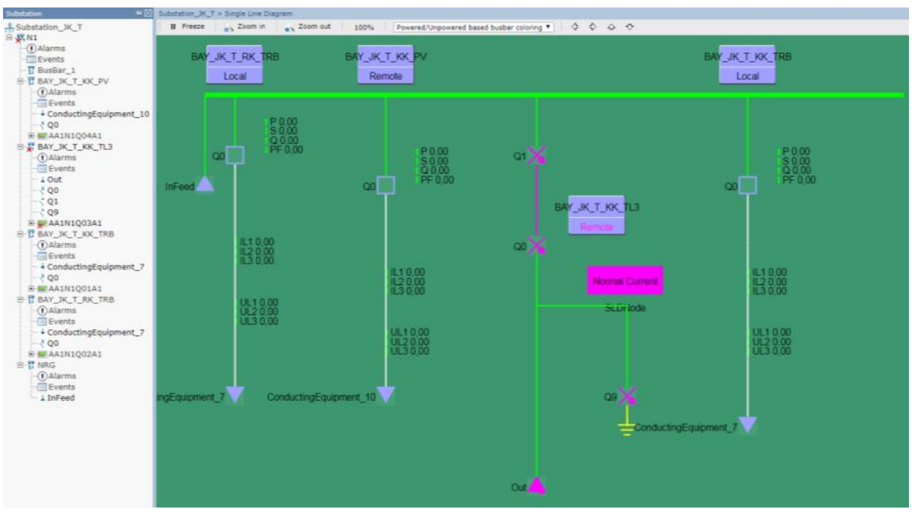

Figure 6.6. COM600 Substation structure and Single Line Diagram of the implementation.

After configuring OPC Server, it was possible to add OPC Client to the Gateway. The OPC Client structure was built similar way as the OPC Server by adding objects to the object tree. To build the object tree, the objects were added in the following order: OPC Client object, Channel object and Device object. Device object's Internet IP address and Station address should be modified if necessary, as the IP address should refer to the IP address of the NCC. Data objects do not need to be added similarly than to OPC Server but linked via Cross-Reference function from OPC Server to OPC Client. (ABB 2017, 51–55, 66–72.)

After opening the Cross-Reference function tool, the data objects could be added by dragging them from the OPC Server to the tool. Data objects could be drag individually or simultaneously within a group from an IED, a channel or a server. Data objects dragged to the Cross-Reference tool could be connected to OPC Client slave. The address and other properties of data objects could be also configured, while they were in the Cross-Reference tool. (ABB 2017, 66–72.)

The data objects initially chosen for the Cross-Reference function, moreover for the data transfer were power, current and voltage measurements measured by each REF615. Chosen power measurements were a real power, apparent power, reactive power and power factor. The other chosen measurements were current and phase voltage of each phase. Additionally, a control signal for circuit breaker position was chosen for the data transfer. The measurements and controls are presented in more detail in Appendix B.

OPC Client, which connects to the OPC Server in COM600, is part of the client structure for communicating with NCCs. The entire client consists of OPC Client, Slave/Server Protocol Stack, configuration SCL, OPC Data Access Server and Alarm & Event server components for diagnostics and control. The other main part of the client apart from OPC Client is the Slave Protocol Stack, which connects to the NCCs. Configuration SCL is used for handling OPC Client and Slave Protocol Stack. Additionally, COM600 utilizes 61850-6 SCL and 61850-7 communication modelling regardless of the chosen Slave Protocol Stack. (ABB 2017, 18–19.)

While adding OPC Client the Slave Protocol Stack had been selected as the OPC Client was added by selecting the required Slave Protocol Stack. The Slave Protocol Stacks supported by COM600 are IEC 61850-8-1, IEC 60870-5-104, DNP 3.0 LAN/WAN, Modbus and OPC (ABB 2017, 18–19). In this implementation, IEC 60870-5-104 Slave OPC Client was chosen as the Slave Protocol Stack because of its simple deployment process. IEC 60870-5-104 Slave Client allows NCCs communicating with 60870-5-104 protocol to connect to the slave client for receiving data from COM600 and delivering control signals to the devices' part of the electrical process. COM600 supports both data communication via Ethernet and via serial interfaces, such as RS 232 and RS 485 (ABB 2017, 35–36). IEC 60870-5-104 Slave Client uses TCP/IP communication through an Ethernet interface.

In the OPC Client structure, several properties needed to be configured to establish a connection to NCC. The most important properties were local address, internet access and station address. The local address is a parameter of IEC 104 Channel under Communication port and IP address of the COM600, which is used for communication, was added to this parameter. Internet access and station address could be found under IEC 104 IED Addresses. IP address of the master system was added to the Internet access, which is IP address of Redis RTDB in this implementation. Station address is the common address of Application Service Data Unit (ASDU) in an IEC 60870 message. Table 6.1 presents the values of these properties.

Table 6.1. Configured properties of IEC 60870-5-104 Slave OPC Client.

Property	Value
Local Address	10.0.0.11
Internet Access	10.0.0.2
Station Address	1

After configuring the IEC 60870-5-104 Slave OPC Client, it was possible to add the required data attributes and objects to the OPC Client via Cross-Reference function. To establish the connection between OPC Client and OPC Slave for each data object, an indication address was added for measurements, while a command address and an indication address for control signals. Additionally, the properties Send As Measurand As Value Type and Time Tag Handling were modified. Send As Measurand As Value Type was changed to Send with float value and Time Tag Handling was changed to Send Long Format Time Tag.

6.1.2 Real-time database

Redis was installed by downloading Redis-x64-3.0.504.zip zip-file by following the link on a redis.io/download web page from MicrosoftArchive/redis on GitHub (Microsoft Archive 2017). Since Redis does not officially support Windows, the downloading of zip file was not possible from the Redis web page itself. After downloading the zip file and extracting its content, the testing of whether the installation was successful was conducted by opening the redis-cli file of the extracted files and executing command ping, which returned PONG as a sign of successful connection between Redis Client and Server. At first, running Redis Server was not possible because of lack of available space in the heap memory in the system-paging file. Therefore, the computer was restarted, as restarting defragments the system-pacing file to allow Redis operation. After the restart, the installation was successful according to the testing.

Local Redis was connected to the cloud Redis via replication, for which the "redisRepl" replication program was downloaded from Daniele Pala / redisRepl on Bitbucket (Pala 2017a). Additionally, security certificates were needed to establish the connection to the cloud Redis. All three required security certificates were provided by RSE SpA. After adding the replication program and the security certificates to the same directory with local Redis client and server, the interconnection could be established. The replication command consisted of four parts, which were security certificates, local Redis IP address, the namespace for measurements and commands and a remote endpoint. For receiving data from the cloud Redis, the replication command could be modified. It was also possible to replicate local Redis' of other ERIGrid partners by modifying the replication command. After establishing the interconnection, all the values updated were replicated to the cloud Redis until the connection is disconnected. (Pala 2017b.)

Redis could be accessed via the command prompt with rather simple commands. The main commands used were `SET` and `GET`. `SET` command sets `key` to hold a specified `value` string, while `GET` command displays the `value` of the requested `key`. In this implementation, only simple keys, which assign single key a single string value, were used, although Redis supports several other data types also, such as lists, sets and hashes. (Redis 2017.)

6.2 Programming

The data objects chosen to be added to the Cross-Reference function of the COM600 were mapped and named according to the naming convention of ERIGrid. Within this naming convention, the name of a data object consists of two parts, which are object name and signal name. The object name describes the subsystem the data object is located and therefore follows the naming convention of object type-system-subsystem. In the object name, a single dot is used to separate hierarchal levels and an underscore is used as a "hierarchy break" marker. For instance, an object name could be

Protection_JK-T.KK-TL3.REF615,

in which the Protection stands for protection device, JK-T the substation, KK-TL3 the subsystem and REF615 is the name of the actual physical device. The second part of the data object name is the signal name. The signal naming follows the order of signal type, domain, signal and subsignal. For instance, a signal name could be

M_EA_TotW.instMag,

in which M stands for measurement type, EA for electrical AC domain, TotW is the signal name for total power and instMag the signal attribute for the magnitude of an instantaneous value. To build a data object name, the signal name is added after the object name, while the names are separated by a colon.

For instance, a data object name for magnitude of total power measured by ABB REF615 relay could be

Protection_JK-T.KK-TL3.REF615:M_EA_TotW.instMag.

(DTU 2017, 51–60.)

The simplest method to connect to COM600 was by connecting to the IEC 60870-5-104 Slave OPC Client with a master based on the IEC 60870-5-104 protocol. Therefore, the connection to the COM600 was conducted with a library based on IEC 60870-5 protocol called lib60870, which has implementations for both client and server (MZ Automation GmbH 2017). This library has two versions lib60870-C based on C99 standard and lib60870.NET, which is a .NET implementation in pure C# and can be operated on Microsoft Visual Studio. In this implementation, the lib60870.NET version was used.

At first, Microsoft Visual Studio 2017 Community version was downloaded from the Visual Studio webpage. Then lib60870 library was downloaded from the GitHub page mz-automation / lib60870 (MZ Automation GmbH 2017). The lib60870.NET solution was opened within Visual Studio IDE (Integrated Development Environment), on which it was possible to test and run the nine example projects provided by the library. The examples include multiple clients and servers for different purposes. One of the simplest example clients was used as a base for the program connecting COM600 and Redis.

The example client program used as a base for the IEC 104 client handle included commands for connecting to the IEC 104 Slave, a connection handler function, a handler function for received ASDUs and commands for closing the connection to the IEC 104 Slave. The example client also contained functions for sending interrogation commands and for sending control commands. Sending interrogation commands was executed with the following command

```
con.SendInterrogrationCommand (CauseOfTransmission.
ACTIVATION, 1, 20);
```

where 1 indicates the common address and 20 indicates station interrogation for requesting all the data points. Sending control commands was executed with the following command

```
con.SendControlCommand  (TypeID.C_SC_NA_1,  CauseOfTransmission.
ACTIVATION, 1, new SingleCommand (5000, true, false, 0));
```

where `TypeID` indicates the command type, 5000 indicates the information object address, `true` is the Boolean command, `false` is a value of `selectCommand` parameter and 0 indicates the qualifier.

The example client was modified by adding to the `asduReceivedHandler` function else-if structures for handling other ASDU Type IDs, which were used by the IEC 104 Slave, but were not included in the simple example. The `asduReceivedHandler` was also modified by adding a new argument for passing the connection variable to the function. Since `asduReceivedHandler` is derived from the lib60870 library, the library's `Connections` file was modified to include the new argument also. Lastly, the program was modified by inserting the sending of interrogation and control command functions to a while loop to enable the program run continuously several times.

To access local Redis via the Visual Studio program, a Redis client was required. In this implementation, the Redis client used was Stack.Exchange.Redis (Stack Exchange 2017). The Redis client was installed in the Visual Studio through NuGet Package Management Package Manager Console with the following command

```
PM> Install -Package StackExchange.Redis
```

The Redis client provides commands for accessing Redis server from the Visual Studio C# program. Connection to the Redis server and the database can be executed with the following lines

```
ConnectionMultiplexer  redisConn  =  ConnectionMultipleXer.Connect
(“localhost”);

IDatabse redDb = redisConn.GetDatabase();
```

The first line connects the `redisConn` variable to the Redis server within the local host and the second line established a connection to a database on the Redis server, to which the `redisConn` was connected. Furthermore, this database can be accessed with the `redDb` variable. Information can be set to the Redis server with the following command

```
redisDb.StringSet("key", "value");
```

The `StringSet` command operates similar to the Redis command prompt command `SET` and sets to a specified key to the given value in the specified database `redisDb`. The control signals from cloud Redis can be acquired with the following command

```
string value = redDb.StringGet("key");
```

The `StringGet` command operates similar to the Redis command prompt command `GET` and gets the requested value from the specified database `redDb`, which is then assigned to a variable `value` to further transfer the control signal to the COM600. The whole program is presented in Appendix C.

6.3 Testing

At first, the connection between the local Redis and the cloud Redis was tested with a simple data transfer task. The replication was started with the replication command from the command prompt of the local Redis. After establishing the replication connection, the signal mapping tags in Appendix B were assigned random values and set to the local Redis as simple keys. At the command prompt, it was possible to follow the replication process based on which the test was successful from the perspective of the local Redis. Later, the test was confirmed successful from the cloud Redis side also by RSE SpA.

Additionally, testing of the delays in the data transfer was conducted. This testing was completed in three sections, which were data transfer between VTT's local Redis and cloud Redis, between RSE's local Redis and VTT's local Redis, and between ABB REF615 relay and VTT's local Redis. The objective of the delay testing was to obtain example values of possible delays to realise the effects of the possible delays for future ERIGrid test cases. However, it should be noted that the result delays were not statistical and did not represent an average value for a delay. The testing was conducted by comparing timestamps. In the test between VTT's local Redis and cloud Redis, a command for sending a timestamp from the local Redis was created, while both the timestamps were gathered from cloud Redis, which automatically displayed the timestamp of the data received. Testing between RSE's local Redis and VTT's local Redis was conducted similarly by RSE adding command for sending a control signal and the corresponding timestamp to VTT's local Redis, while both of the timestamps were gathered from VTT's local Redis, which automatically displayed the timestamps for the data received. The testing between REF615 and VTT's local Redis was conducted by creating simple test setup at the laboratory, which is presented in Figure 6.7. The test setup included a direct connection to the distribution grid, from where electricity was fed through a busbar and two REF615 relays containing one breaker each to the load, a 9 kW heating fan by Malmbergs, which was connected to the connection point for external devices. Based on this test setup, REF615 sent voltage, current and power measurement data with timestamps to the local Redis, where a second timestamp was created for the data received.

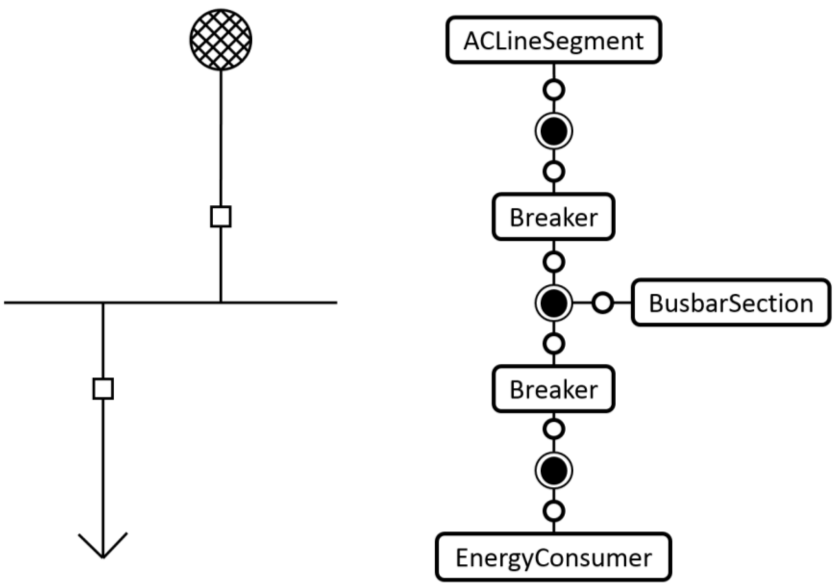

Figure 6.7. Traditional single line diagram and CIM representation of the test set up at the Multipower laboratory.

All the timestamps obtained during this testing were in UTC (Coordinated Universal Time). The timestamps sent from VTT’s local Redis to the cloud Redis and from RSE’s local Redis to VTT’s local Redis were in Unix time, while rest of the timestamps were presented according to UTC date and time. None of the equipment used was time synchronized. The local Redis databases at both VTT and RSE received time from the Internet; in particular, VTT’s local Redis received time from NIST (National Institute of Standards and Technology) Internet time service. REF615 received time from GPS system at the laboratory. Caused by the lack of time synchronization, the comparison between timestamps was inaccurate but was still able to provide rough estimates for possible delays. Additionally, the delays between Redis databases should be considered quite optimistic, since only small amount of data was transferred.

A major challenge for the testing was the time synchronization as the initial tests failed due to the local Redis of VTT using local time from the computer, which was roughly 1.5 minutes slow in comparison to the UTC(NIST). This issue was partially solved by retrieving time from NIST, though this solution was still not as accurate as implementing time synchronization would have been.

The example delays for data transfer between VTT’s local Redis and cloud Redis ranging from 184 milliseconds to 600 milliseconds. Most of the values for the delay were between 200 milliseconds and 400 milliseconds. The example delay between RSE’s local Redis and VTT’s local Redis was 1 minute 28 seconds. Lastly, the example delays between REF615 and VTT’s local Redis were 7 seconds and 33 seconds. Detailed results of the delay testing are presented in Appendix D. A brief analysis of the results is provided in Chapter 7.

7 DISCUSSION

This chapter discusses the results of the testing described in Section 6.3. Moreover, this chapter provides a brief analysis of the testing results and the research questions presented in Section 1.2. The overall implementation of the real-time laboratory interconnection was successful. Data transfer to both directions worked correctly between local Redis and cloud Redis, and between local Redis and COM600. The transferred data also included quality indicators and timestamps.

There are already available several different interconnection and remote connection implementations, which were conducted with various ICT communication technologies. Most of the connection implementations utilized TCP/IP technologies. However, some of the implementations were conducted via cloud. JaNDER has also adopted a cloud-based communication implementation as with cloud-based implementation the challenges regarding modifying firewalls at the laboratories could be avoided. Especially, at Multipower laboratory security issues were avoided due to the cloud-based implementation. The laboratory connection at Multipower laboratory was implemented by interconnecting COM600 and local Redis via programs containing an IEC 104 client and a Redis client. The interconnection to the cloud Redis was implemented with a replication program by Pala (2017a). The current implementation seems efficient and conducts the data transfer in a sufficient manner. However, there are improvement possibilities for the current version. The program could be studied thoroughly for possibilities to improve efficiency. The interconnection from COM600 was implemented with IEC 60870-5-104. The possibilities for updating the IEC 104 client to an IEC 61850 client should be considered since rest of the implementation was based on IEC 61850. Additionally, the possibilities of Java implementation should be considered, as the library available in Java is more robust than the library in C#.

There are several cases, in which a real-time laboratory interconnection could be beneficial. The interconnection is beneficial if it is not possible to visit the physical laboratory since the interconnection can be accessed remotely via the Internet. The number of equipment available for testing is also increased due to the interconnection, which could be beneficial for laboratories and researchers with less equipment available at their location. In addition, complex test cases could be implemented with the interconnection, since there would be more equipment available. A real-time laboratory interconnection could be also used if the efficiency and utilization of the research facilities should be increased, for instance, due to economic reasons. In this case, the interconnection would allow other research groups to access the research facility, while also increasing the possibilities for further collaboration within the consortium and other laboratories implementing the real-time laboratory interconnection.

Based on the benefits of the interconnection, there are multiple beneficiaries. The laboratory, to which the interconnection has been implemented, benefits from the interconnection through the increased efficiency and utilization of the laboratory. Additionally, the researchers at the laboratory can possibly find more opportunities to collaborate with others, since the laboratory is already connected to the common platform. All the research facilities can benefit from the wider pool of equipment available for testing. Since accessing a wider range of equipment is possible via the interconnection, it could be beneficial for industry and other research groups without proper research infrastructure. Industry and other researchers could access several laboratories from one physical laboratory. The interconnection could also provide possibilities for the industry to test their products in a wider test environment and gain references for their products from a number of research institutes. Overall, benefits, such as the development of technologies based on the complex test cases conducted via the interconnection, can contribute to advancements in power systems and policymaking, while also benefitting society by maintaining reliability of the power system in future. Therefore, DSOs, TSOs, electricity retailers, policymakers and end users could also benefit from implementing and testing interconnections and ICT communication on power grids.

There can be seen huge potential for implementing interconnections to all types of power system laboratories. The potential laboratories to apply an interconnection should have data, which they could share via the interconnection, for instance, measurements and control signals. The potential laboratory should also have the equipment, which they could connect via the interconnection to the platform. This equipment should have at least feature to collect measurement data and possibly also feature to support remote control for receiving control signals through the interconnection. However, the equipment would not need to be greatly advanced, as power grid components with a simple ICT communication would be sufficient also. It would be preferred to connect research facilities, which have already implemented a common ICT communication standard within the facility, such as IEC 61850 or CIM. However, JaNDER level 0 implementation has highlighted that the interconnection is possible to implement even without common ICT communication standard and technologies if all the partners follow the same naming convention and gather all the data first to a local database and build the interconnection from the local database only.

Although there are several potential research facilities to implement an interconnection, there are also many challenges, which should be first considered. A major challenge for the interconnections is security, which should be considered from perspectives of remote control and cyber-attacks. Remote control should be designed and implemented with processes checking the feasibility of a given control signals. Solutions for disconnecting the interconnection or allowing the facility to connect only for certain periods of time could be considered in addition to safety procedures to avoid cyber-attacks. Additionally, a challenge could occur with research facilities unwilling to adopt different communication standards and technologies to unify their facilities with others, when the researcher groups of these facilities are competing in the same market. Moreover, as laboratories would not need to invest to the same equipment since the equipment is available via the interconnection, policies might be needed to ensure that each nation or region has state-of-art equipment available within its research facilities. Funding and investments could also create tension if the physical components and equipment of the research facilities were spread across two regions. It is clear that there are facilities, which could not implement the interconnection, for instance, because of research for military purposes or high-security clearance with all data of the laboratory confidential.

The example delays results from the testing described in Section 6.3, can be considered quite optimistic since only small amount of data was transferred. According to Kurose & Ross (2013, 37), a delay could last up to several hundred milliseconds, thus the example delay results fit this range. According to Coffele et al. (2010, 5), a maximum value for the delay between IED and COM600 was assessed as 100 milliseconds. The longer delays during the last two testing situations could be due to the non-existent time synchronization, queuing and transmission delays, using one-way measurement of timestamps in the testing, and due to retrieving time from the Internet, which has lower consistency and predictability than for instance cable connections (Kurose et al. 2013, 37–41; Kay 2009, 24, 27–29). Additionally, the laboratory computer was a simple basic laptop and the routers and modems used to transfer the data between VTT and RSE were unknown, thus the laptop, routers or modems used could have become overloaded.

The testing could be improved by increasing the number of observation points, to gain a better understanding in which parts of the data transfer the delays occur. The tests could be conducted as a round trip measurement since round-trip measurement testing does not require time synchronization as the data will be sent and received at the same location. However, if one-way testing is conducted, GPS could be a potential method to time synchronize the test setup since GPS has better accuracy than retrieving time from the Internet. (Kay 2009, 24, 27–29.)

8 CONCLUSIONS

The decentralized energy production in electrical power systems is increasing as a result of the mitigation and adaptation to anthropogenic global warming by deploying renewable energy. One of the best techno-economical solutions for future renewable energy system is the smart grid. Smart grids need advanced automation and ICT communication technologies to maintain the level of reliability and power quality present in traditional centralized power systems.

In this thesis, a real-time laboratory interconnection was implemented at the Multipower laboratory. The implementation was successful as data was transferred via the real-time laboratory interconnection to both directions. The implementation conducted in this thesis was limited to the JaNDER level 0 implementation. The real-time laboratory interconnection was implemented by interconnection a substation automation and management unit, ABB COM600, with local Redis RTDB, which was further interconnected via a replication program, redisRepl, to the cloud-based Redis RTDB. Libraries StackExchange.Redis and lib60870 were used in the program written on Visual Studio C# to interconnect COM600 and local Redis. The work conducted in this thesis was based on IEC 61850, CIM and IEC 60870-5-104.

There are several different possibilities for implementing an interconnection or a remote connection to a laboratory or a DER unit. The most relevant implementation possibilities were discussed in the literature review. Most of the implementations discussed were conducted based on TCP/IP model because the model was implemented based on the most used communication technologies. A modern competitor for TCP/IP technologies was found to be cloud-based technologies, which use cloud computing in establishing the interconnection and data transfer. The JaNDER implementation conducted in this thesis was cloud-based, while the implementation within Multipower laboratory was based on TCP/IP and Ethernet.

The key benefits of a real-time laboratory interconnection are an increase in available equipment, possibilities for complex test cases and an increase in collaboration and in utilization of laboratories. The main beneficiaries derived from the benefits are the research facility and its researchers, consortium partners and research groups without their own research facilities, for whom it is possible to access an increased amount of equipment via a single research facility. Additionally, a real-time interconnection can be seen beneficial for society at large because of the enhanced power system development possibilities.

The research facilities suitable for implementing a real-time laboratory interconnection should have data, for instance, measurements and control signals, which could be shared via the interconnection. The research facilities should also have the equipment, which could be connected via the interconnection. However, remote control of any equipment, which could cause security threads at the physical location, should not be interconnected. Additionally, it might be easier to implement a real-time laboratory interconnection at research facilities, which have already implemented ICT communication based on a specific standard. Although, JaNDER level 0 implementation has highlighted that it is possible to implement the interconnection without foundation on a specific common standard, but rather following common guidelines and naming conventions.

The results obtained in the testing portray possible example values for delays; neither statistical nor average values nor was the testing equipment time synchronized. The example delays from VTT’s local Redis to cloud Redis were between 184 and 600 milliseconds, an example delay from RSE’s local Redis to VTT’s local Redis was 1 minute 28 seconds, while example delays from ABB REF615 to VTT’s local Redis were 7 seconds and 33 seconds. The example delays from VTT’s local Redis to cloud Redis fit to the range discussed in the literature. Longer delays could be due to several reasons, for instance, lack of time synchronization, queuing and transmission delays, using the one-way measurement of timestamps in the testing, and due to retrieving time from the Internet.

In further research, updating the IEC 60870-5-104 library to IEC 61805 library in Java should be considered. Within the ERIGrid project, the JaNDER level 1 and level 2 implementations and the test cases should be conducted. Additionally, profound delay testing with time synchronization should be conducted at Multipower laboratory to obtain statistical delays and energy management system should be established to be able to add more equipment measurements and control signals to the Redis RTDB.

REFERENCES

ABB. 2017. COM600 series, version 5.0 User’s Manual [online document]. [Accessed 14 July 2017]. Available at https://search-ext.abb.com/library/Download.aspx?DocumentID=1MRS756125&LanguageCode=en&DocumentPartId=&Action=Launch

ABB. 2008. When Grids Get Smart ABB’s Vision for the Smart Grid [online document]. [Accessed 19 July 2017]. Available at http://www02.abb.com/db/db0003/db002698.nsf/0/23bd705661c12f6ec12575bb002a5be9/$file/ABB+paper+smart+grids+June+version.pdf

AIT (Austrian Institute of Technology GMBH) et al. 2014. H2020-INFRAIA-2014/2015 - European Research Infrastructure supporting Smart Grid Systems Technology Development, Validation and Roll Out (ERIGrid) - Part B-I (Technical Annex Section 1–3). Research and Innovation Actions (RIA) Proposal. European Commission.

Alani, M. M. 2014. Guide to OSI and TCP/IP Models. Cham: Springer.

Apostolov, A. P. 2010. UML and XML use in IEC 61850 [online document]. In Proceedings of Transmission and Distribution Conference and Exposition 2010 IEEE PES, April 19–22, New Orleans, LA, USA. pp. 1–6. [Accessed 6 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5484325&isnumber=5484192

Baigent, D., Adamiak, M. & Mackiewicz, R. 2004. IEC 61850 Communication Networks and Systems in Substations: An Overview for Users [online document]. In Proceedings of *the VIII simposio iberoamericano sobre proteccion de sistemas electricos de potencia*, Monterey, Mexico: Citeseer. pp. 1–15. [Accessed 20 June 2017]. Available at: https://pdfs.semanticscholar.org/8339/cfc685edef6a4a0955e7c14c30bb1bcec19e.pdf

Belgradskaya, E. V. & Ignatiev, K. V. 2014. Методические указания для проведения лабораторных работ удаленного доступа [Methodical instructions for conducting remote access laboratory work]. Saint Petersburg: Saint Petersburg Electrotechnical University “LETI”, Faculty of

Industrial Automation and Electrical Engineering, Department of Automatic Control Systems.

Besada-Portas, E., Lopez-Orozco, J.A., De La Torre, L. & De La Cruz, J.M. 2013. Remote control laboratory using EJS applets and TwinCAT programmable logic controllers [online document]. IEEE Transactions on Education, vol. 56, no. 2, pp. 156–164. [Accessed 27 June 2017]. Available at: http://ieeexplore.ieee.org/document/6220875/

Buchmann, A. 2005. Real Time Database Systems. Encyclopedia of Database Technologies and Applications. Ed. Laura C. Rivero, Jorge H. Doorn, and Viviana E. Ferraggine. Idea Group.

Cinergia. 2016. GE-AC Grid Emulator [online document]. [Accessed 23 May 2017]. Available at http://www.cinergia.coop/sites/default/files/fitxage-ac_v3.pdf

Coffele, F., Moore, P., Booth, C., Dysko, A., Burt, G., Spearing, T. & Dolan, P. 2010. Centralised Loss of Mains protection using IEC-61850. In Proceedings of 10th IET International Conference on Developments in Power System Protection (DPSP 2010), March 29 - April 1, Manchester, United Kingdom. pp. 1–5. [Accessed 23 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5522232&isnumber=5522091

Commission Staff Working Document Definition, Expected Services, Functionalities And Benefits Of Smart Grids Accompanying documents to Communication From The Commission To The European Parliament, The Council, The European Economic And Social Committee And The Committee Of The Regions Smart Grids: from innovation to deployment, COM(2011) 202, final

CORDIS (Community Research and Development Information Service). 2014. Final Report Summary - DERri (Distributed Energy Resources Research Infrastructure). Report Summary. [Accessed 20 June 2017]. Available at http://cordis.europa.eu/result/rcn/153744_en.html

Della Giustina, D., Andersson, L. & Ravera Iglesias, G. J. 2013. Experimental Performance Characterization of a Meshed Network for the Smart Grid [online document]. In Proceedings of 2013 4th IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe), October 6–9, Lyngby, Denmark. pp. 1–5. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/6695415/

Della Giustina, D., Repo, S., Zanini, S., Cremaschini, L., 2011. ICT architecture for an integrated distribution network monitoring [online document]. In Proceedings of 2011 IEEE International Workshop on Applied Measurements for Power Systems (AMPS), September 28–30, Aachen, Germany. pp. 102–106. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/6090435/

DERlab (European Distributed Energy Resources Laboratories) e.V. 2013. Activity Report 2012/2013 [online document]. [Accessed 20 June 2017]. Available at http://der-lab.net/wp-content/uploads/2016/05/derlab_activity_report_2012_2013_LQ.pdf

DERri (Distributed Energy Resources Research Infrastructure). 2013a. JRA1 - Joint Test Facility for Smart Energy Networks with Distributed Energy Resources JaNDER. [www-page]. [Accessed 20 June 2017]. Available at http://www.der-ri.net/index.php?id=22

DERri. 2013b. Test Facilities Interoperation [online document]. [Accessed 20 June 2017]. Available at http://www.der-ri.net/uploads/media/Test_Facilities_Interoperation.pdf

DERri. 2012. Deliverable D1.3, “Definition of the interfaces: Requirements specification,” FP7 DERri project.

DERri. 2010. Deriverable D1.1, “Detailed description of JaNDER concept,” FP7 DERri project.

Directive 2009/72/EC of the European Parliament and of the Council of 13 July 2009 concerning common rules for the internal market in electricity, *Official Journal* L 211.

Directive 2012/27/EU of the European Parliament and of the Council of 25 October on energy efficiency, *Official Journal* L 315/1.

DLMS User Association. 2007. DLMS/COSEM Architecture and Protocols Excerpt from Companion Specification for Energy Metering [online document]. [Accessed 8 November 2017]. Available at: http://dlms.com/documents/archive/Excerpt_GB6.pdf

DTU (Danmarks Tekniske Universitet) et al. 2017. Deliverable D-JRA3.1,"Improved hardware and software interfaces," H2020 ERIGrid project.

Elgargouri, A., Elfituri, M. M. & Elmusrati, M. 2013. IEC 61850 and Smart Grids [online document]. In Proceedings of 2013 3rd International Conference on Electric Power and Energy Conversion Systems, October 2–4, Istanbul, Turkey. pp. 1–6. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/6713080/

Elgargouri, A., Virrankoski, R. & Elmusrati, M. 2015. IEC 61850 based smart grid security [online document]. In Proceedings of 2015 IEEE International Conference on Industrial Technology (ICIT), March 17–19, Seville, Spain. pp. 2461–2465. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/7125460/

EPRI (Electric Power Research Institute). 2015. Common Information Model Primer. Technical report. 3rd Edition. Palo Alto, CA.

ERIGrid. 2016. Transnational Access Provision - Multipower Laboratory VTT Technical Research Centre of Finland. Research Infrastructure Description and Transnational Access Conditions. European Commission.

Erl, T., Mahmood, Z. & Puttini, R. 2013. Cloud Computing - Concepts, Technology & Architecture. Westford, MA: Prentice Hall.

Gallagher, P. D. 2014. NIST Framework and Roadmap for Smart Grid Interoperability Standards [online document]. Release 3.0, Technical Report Draft, NIST (National Institute of Standards and Technology). [Accessed 16 November 2017]. Available at: https://www.nist.gov/sites/default/files/documents/smartgrid/Draft-NIST-SG-Framework-3.pdf

Gehrke, O. 2017. Enhanced lab-based testing methods and tools. Center for Electric Power and Energy, Technical University of Denmark. In Proceedings of Designing and Validating the Future Intelligent, Electric Power Systems Workshop, September 6, Kassel, Germany.

Güngör, V.C., Sahin, D., Kocak, T., Ergüt, S., Buccella, C., Cecati, C. & Hancke, G.P. 2011. Smart grid technologies: Communication technologies and standards [online document]. IEEE Transactions on Industrial Informatics, vol. 7, no. 4, pp. 529–539. [Accessed 26 June 2017]. Available at: http://ieeexplore.ieee.org/document/6011696/

Hargreaves, N. B., Pantea, S. M. & Taylor, G. A. 2014. Adopting the IEC Common Information Model to Enable Smart Grid Interoperability and Knowledge Representation Processes. In: Hossain, J. & Mahmud, A. eds. *Large Scale Renewable Power Generation*. Singapore: Springer. pp. 439–462.

Hegedus, A. S., Copot, C., Chevalier, A., Nascu, I. & De Keyser, R. 2013. Remote laboratory: A novel tool for control engineering laboratories [online document]. In Proceedings of 2013 17th International Conference on System Theory, Control and Computing (ICSTCC), October 11–13, Sinaia, Romania. pp. 256–261. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/6688969/

International Electrotechnical Commission (IEC). 2013. IEC/TR 61850-1: 2013. Communication networks and systems for power utility automation - Part 1: Introduction and overview. ed2.0. IEC.

International Electrotechnical Commission (IEC). 2011. IEC 61850-7-1: 2011. Communication networks and systems for power utility automation - Part 7-1: Basic communication structure - Principles and models. ed2.0. IEC.

International Electrotechnical Commission (IEC). 2010. IEC/TR 61850-7-2: 2010. Communication networks and systems for power utility automation - Part 7-2: Basic information and communication structure - Abstract communication service interface (ACSI). ed2.0. IEC.

IPCC (Intergovernmental Panel on Climate Change). 2014. *Climate Change 2014: Synthesis Report*. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Core Writing Team, R.K. Pachauri and L.A. Meyer (eds.)]. IPCC, Geneva, Switzerland.

Islam, S.N., Mahmud, M.A. & Oo, A.M.T. 2016. Relay aided smart meter to smart meter communication in a microgrid [online document]. In Proceedings of 2016 IEEE International Conference on Smart Grid Communications, (SmartGridComm). November 6–9, Sydney, Australia. pp. 128–133. [Accessed 26 June 2017]. Available at: http://ieeexplore.ieee.org/document/7778750/

International Standardization Organization (OSI). 1994. ISO/IEC 7498-1:1994. Information technology - Open Systems Interconnection - Basic reference model: The basic model. ed2.0. OSI.

Ivanov, C. 2016. The Way to Exchange: What Is the Common Information Model [online document]? [Guest Editorial]. IEEE Power and Energy Magazine, vol. 14, no. 1, p. 22–28. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/7366382/

Jacobsen, R.H. & Mikkelsen, S.A. 2014. Infrastructure for intelligent automation services in the smart grid [online document]. Wireless Personal Communications, vol. 76, no. 2, pp. 125–147. [Accessed 22 June 2017]. Available at: https://link.springer.com/article/10.1007/s11277-014-1682-6

Kasztenny, B., Whatley, J., Urden, E. A., Burger, J., Finney, D. & Adamiak, M. 2006. IEC 61850 - A Practical Application Primer for Protection Engineers [online document]. In Proceedings of 2006 Power Systems Advanced Metering, Protection, Control, Communication, and Distributed Resources, March 14–17, Clemson, SC, USA. pp. 18–50. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/4064649/

Kay, R. 2009. Pragmatic Network Latency Engineering Fundamental Facts and Analysis. cPacket Networks Inc. pp. 1–31. [Accessed 24 November 2017]. Available at: https://pdfs.semanticscholar.org/a19d/53a533fe78d01f0423f628e4d1688be23e6d.pdf

Khan, A.A., Rehmani, M.H. & Reisslein, M. 2016. Cognitive radio for smart grids: Survey of architectures, spectrum sensing mechanisms, and networking protocols [online document]. IEEE Communications Surveys and Tutorials, vol. 18, no. 1, pp. 860–898. [Accessed 26 June 2017]. Available at: http://ieeexplore.ieee.org/document/7275088/

Kheaksong, A. & Lee, W. 2014. Packet transfer of DLMS/COSEM standards for smart grid [online document]. In Proceeding of 2014 Asia-Pacific Conference on Communications (APCC), October 1–3, Pattaya, Thailand. pp. 391–396. [Accessed 8 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7092843&isnumber=7091593

Kolenc, M., Nemček, P., Gutschi, C., Suljanović, N. & Zajc, M. 2017. Performance evaluation of a virtual power plant communication system providing ancillary services [online document]. Electric Power Systems Research, vol. 149, pp. 46–54. [Accessed 22 June 2017]. Available at: http://www.sciencedirect.com/science/article/pii/S0378779617301578

Kurose, J. F. & Ross, K. W. 2013. Computer networking: a top-down approach. 6th ed. Pearson Education, Inc.

Lakervi, E. & Partanen, J. 2008. Sähkönjakelutekniikka [Electricity Distribution Technology]. Helsinki: Otatieto / Gaudeamus Helsinki University Press.

Lee, B. & Kim, D. 2017. Harmonizing IEC 61850 and CIM for connectivity of substation automation [online document]. Computer Standards & Interfaces, vol. 50, pp. 199–208. [Accessed 17 November 2017]. Available at: http://www.sciencedirect.com/science/article/pii/S0920548916301295

Leksawat, S., Schmelter, A., Ortjohann, E., Holtschulte, D., Kortenbruck, J. & Morton, D. 2015. Implementation of communication model and web services for cluster-based power system operation in smart grids [online document]. In Proceedings of the 2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA), November 3–6, Bangkok, Thailand. pp. 1–6. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/7387039/

León, H., Montez, C., Stemmer, M. & Vasques, F. 2016. Simulation models for IEC 61850 communication in electrical substations using GOOSE and SMV time-critical messages [online document]. In Proceedings of 2016 IEEE World Conference on Factory Communication Systems (WFCS), May 3–6, Aveiro, Portugal. pp. 1–8. [Accessed 14 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7496500&isnumber=7496486

Mackiewicz, R. E. 2006. Overview of IEC 61850 and Benefits [online document]. In Proceedings of 2006 IEEE PES Power Systems Conference and Exposition, October 29 – November 1, Atlanta, GA, USA. pp. 623–630. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/4075831/

Martini L., Radaelli L., Brunner H., Caerts C., Morch A., Hanninen S. & Tornelli C. 2015. ELECTRA IRP Approach to Voltage and Frequency Control for Future Power Systems with High DER Penetration [online document]. In Proceedings of 23rd International Conference on Electricity Distribution, June 15–18, Lyon, France. pp. 1–5. [Accessed 6 November 2017]. Available at: http://cired.net/publications/cired2015/papers/CIRED2015_1357_final.pdf

McMorran, A. W., Lincoln, R. W., Taylor, G. A. & Stewart, E. M. 2011. Addressing Misconceptions about the Common Information Model (CIM) [online document]. In Proceedings of 2011 IEEE Power and Energy Society General Meeting, July 24–29, Detroit, MI, USA. pp. 1–4. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/6039391/

Messinis, G., Kleftakis, V., Kouveliotis-Lysikatos, I., Rigas, A., Vassilakis, A., Kotsampopoulos, P. & Hatziargyriou, N. 2014. A multi-microgrid laboratory infrastructure for smart grid applications [online document]. In Proceedings of MedPower 2014, November 2–5, Athens, Greece. pp. 31–37. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/7386111/

Mercurio, A., Di Giorgio, A. & Cioci, P. 2009. Open-Source Implementation of Monitoring and Controlling Services for EMS/SCADA Systems by Means of Web Services - IEC 61850 and IEC 61970 Standards [online document]. IEEE Transactions on Power Delivery, vol. 24, no. 3, pp. 1148–1153. [Accessed 17 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5109844&isnumber=5109837

Merino J., Rodriguez E., Caerts C., Visscher K., D'hulst R., Rikos E. & Temiz A. 2015. Scenarios and requirements for the operation of the 2030+ electricity network [online document]. In Proceedings of 23rd International Conference on Electricity Distribution, June 15–18, Lyon, France. pp. 1–5. [Accessed 6 November 2017]. Available at: http://cired.net/publications/cired2015/papers/CIRED2015_1329_final.pdf

Microsoft Archive. 2017. Redis: Releases [Program]. [Accessed 27 November 2017]. Available at: https://github.com/MicrosoftArchive/redis/releases

Mostefaoui, H., Benachenhou, A. & Benattia, A.A. 2017. Design of a low cost remote electronic laboratory suitable for low bandwidth connection [online document]. Computer Applications in Engineering Education, vol. 25, no. 3, pp. 480–488. [Accessed 22 June 2017]. Available at: http://onlinelibrary.wiley.com/doi/10.1002/cae.21815/abstract

MZ Automation GmbH. 2017. lib60870 [Program]. [Accessed 27 November 2017]. Available at: https://github.com/mz-automation/lib60870

Mäkelä, M. 2016. Operation of Wind Turbines [PowerPoint presentation]. BL40A2600 Wind Power and Solar Energy Technology and Business. Lappeenranta University of Technology, School of Energy Systems, Electrical Engineering, Control Engineering and Digital Systems module. [Accessed 15 September 2017]. Available at: http://moodle.lut.fi/pluginfile.php/143548/mod_resource/content/1/Lecture%205%20BL40A2600%202016.pdf

Mäki, K. 2017. Multipower Laboratory Configuration. [Diagram].

Mäki, K., Hashmi, M., Farin, J. & Pykälä, M. L. 2011. Modelling and experimental verification of converter drive system for distributed generation [online document]. In Proceedings of IET Conference on Renewable Power Generation (RPG), September 6–8, Edinburg, United Kingdom. pp. 1–6. [Accessed 22 May 2017]. Available at http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6136169&isnumber=6135952

Nguyen, V.H., Tran, Q.T. & Besanger, Y. 2016. SCADA as a service approach for interoperability of micro-grid platforms [online document]. Sustainable Energy, Grids and Networks, vol. 8, pp. 26–36. [Accessed 10 November 2017]. Available at: http://www.sciencedirect.com/science/article/pii/S235246771630056X

Orda, L. D., Bach, J., Pedersen, A. B., Poulsen, B. & Hansen, L. H. 2013. Utilizing a flexibility interface for distributed energy resources through a cloud-based service [online document]. In Proceedings of 2013 IEEE International Conference on Smart Grid Communications (SmartGridComm), October 21–24, Vancouver, BC, Canada. pp. 312–317. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6687976&isnumber=6687920

Pala, D. 2017a. Downloads / redisRepl [program]. [Accessed 19 October 2017]. Available at: https://bitbucket.org/danielePala/redisrepl/downloads/

Pala, D. 2017b. RE: Security certificates for VTT. [private email]. Receiver: Petra Raussi. Sent 19.10.2017, 18.11 o'clock (GMT+0200).

Palensky, P., van der Meer, A., López, C. D., Joseph, A. & Pan, K. 2017. Cosimulation of intelligent power systems: Fundamentals, software architecture, numerics, and coupling [online document]. IEEE Industrial Electronics Magazine, vol. 11, no. 1, pp. 34–50. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/7883974/

Pradeep, Y., Seshuraju, P., Khaparde, S. A., Warrier, V. S. & Cherian, S. 2009. CIM and IEC 61850 Integration Issues: Application to Power Systems [online document]. In Proceedings of 2009 IEEE Power & Energy Society General Meeting, July 26–30, Calgary, AB, Canada. pp. 1–6.

[Accessed 17 November 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5275765&isnumber=5260217

Redis. 2017. An introduction to Redis data types and abstractions. [www-page]. [Accessed 23 October 2017]. Available at https://redis.io/topics/data-types-intro

Sáenz, J., Esquembre, F., Garcia, F.J., de la Torre, L. & Dormido, S. 2016. A new Model for a Remote Connection with Hardware Devices using Javascript [online document]. IFAC-PapersOnLine, vol. 49, no. 6, pp. 133–137. [Accessed 27 June 2017]. Available at: https://ac.els-cdn.com/S2405896316303718/1-s2.0-S2405896316303718-main.pdf?_tid=0a9d07a6-cca9-11e7-955c-00000aacb362&acdnat=1511041386_d816bb39c86f246bab71bdb804b855fb

Sandoval, K. 2016. Living in the Cloud Stack - Understanding SaaS, PaaS, and IaaS APIs [www-page]. [Accessed 13 November 2017]. Available at: https://nordicapis.com/living-in-the-cloud-stack-understanding-saas-paas-and-iaas-apis/

Sandroni, C., Verga, M. & Pala, D. 2017. JRA4: Implementation and Demos of Use Cases/Scenarios in integrated RI [Presentation]. ERIGrid Technical meeting, September 4–5, Kassel, Germany.

Sandroni, C., Verga, M., Lazzari, R., Fantini, M., Sacchi, M. & Prandoni, V., 2016. RSE’s microgrid: A facility for research, development and testing of future distributed generation and microgrid technologies [online document]. AEIT International Annual Conference (AEIT), October 5–7, Capri, Italy. pp. 1–6. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/7892773/

Santodomingo, R., Rohjans, S., Uslar, M., Rodríguez-Mondèjar, J. A. & Sanz-Bobi, M. A. 2013. Facilitating the Automatic Mapping of IEC 61850 Signal and CIM Measurements [online document]. IEEE Transactions on Power Systems, vol. 28, no. 4, pp. 4348–4355. [Accessed 17 November 2017]. Available at:

http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6552228&isnumber=6627990

Selga, J.M., Zaballos, A. & Navarro, J. 2013. Solutions to the computer networking challenges of the distribution smart grid [online document]. IEEE Communications Letters, vol. 17, no. 3, pp. 588–591. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/6459490/

Sendoya-Losada, D, Silva, P, & Waltero, H. 2016. Automatic Wiring System Applied to the Training Module M2CI [online document]. ARPN Journal Of Engineering & Applied Sciences, vol. 11, no. 19, pp. 11503–11513. [Accessed 20 June 2017]. Available at: http://www.arpnjournals.org/jeas/research_papers/rp_2016/jeas_1016_5104.pdf

Smart Grid Coordination Group (SGCG). 2014. SGCG/M490/G_Smart Grid Set of Standards [online document]. Version 3.1. CEN, CENELEC, ETSI. October 31. [Accessed 16 November 2017]. Available at: ftp://ftp.cencenelec.eu/EN/EuropeanStandardization/HotTopics/SmartGrids/SGCG_Standards_Report.pdf

Skare, P., Falk, H., Rice, M. & Winkel, J. 2016. In the Face of Cybersecurity: How the Common Information Model Can Be Used [online document]. IEEE Power and Energy Magazine, vol. 14, no. 1, p. 94–104. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/document/7366675/

Stack Exchange. 2017. StackExchange.Redis [Program]. [Accessed 27 November 2017]. Available at: https://github.com/StackExchange/StackExchange.Redis

Tian, P., Xiao, X., Chen, Y., Jing, T., Huang, X. & Zhi, N. 2017. The key technologies and analysis of research state of microgrid community [online document]. Resources, Conservation and Recycling, vol. 121, pp. 56–63. [Accessed 26 June 2017]. Available at: http://www.sciencedirect.com/science/article/pii/S0921344916300398

Uslar, M., Specht, M., Rohjans, S., Trefke, J. & González, J. M. 2012. The Common Information Model CIM IEC 61968/61970 and 62325 - A practical introduction to the CIM. Volume 66. Berlin: Springer.

VTT et al. 2017. Deliverable 7.3 - D-JRA1.3 Use case implementation plan. ERIGrid Consortium.

Wirepas. 2017. Wirepas Connectivity Overview Product Description [online document]. [Accessed 7 November 2017]. Available at: https://wirepas.com/download/

Wlas, M., Gackowski, M. & Kolbusz, W. 2011. The Ethernet POWERLINK Protocol for smart grids elements integration [online document]. In Proceedings of 2011 IEEE International Symposium on Industrial Electronics, June 27–30, Gdansk, Poland. pp. 2070–2075. [Accessed 22 June 2017]. Available at: http://ieeexplore.ieee.org/document/5984479/?reload=true&arnumber=5984479

Wollenberg, B., Britton, J., Dobrowolski, E., Podmore, R., Resek, J., Scheidt, J., Russell, J., Saxton, T. & Ivanov, C. 2016. A Brief History: The Common Information Model [In My View] [online document]. IEEE Power and Energy Magazine, vol. 14, no. 1. p. 125–126, 128. [Accessed 20 June 2017]. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7366644

Xu, T., Wade, N.S., Davidson, E.M., Taylor, P.C., McArthur S.D.J. & Garlick, W.G. 2011. Case-based reasoning for coordinated voltage control on distribution networks [online document]. Electric Power Systems Research, vol. 81, no. 12, pp. 2088–2098. [Accessed 20 June 2017]. Available at: http://www.sciencedirect.com/science/article/pii/S0378779611001921

Österlund, L., Hunter, K., Demaree, K., Goodrich, M., McMorran, A., Iverson, B. & Kostic, T. 2016. Under the Hood: An Overview of the Common Information Model Data Exchanges [online document]. IEEE Power and Energy Magazine, vol. 14, no. 1, p. 68–82. [Accessed 19 June 2017]. Available at: http://ieeexplore.ieee.org/document/7366710/

APPENDIX A. Smart grid ICT communication technologies and their features

Table 2. Communication technologies for smart grids and their advantages and disadvantages (Khan et al. 2016, 865).

Standard/ Protocol	Advantages	Disadvantages	Suitable SG Architecture Layer	Applications Area in SG
ZigBee	16 channels with 5 MHz of bandwidth in the 2.4 GHz band; Low power usage; Low complexity; Low deployment cost	Limited battery energy supply; Short range; Low data rate; Low processing capabilities; Interference with appliances utilizing license free ISM frequency band, including IEEE 802.11 WLANs, WiFi, Bluetooth and Microwave	HAN; AMI	Smart lighting; Energy monitoring; Home automation; Automatic meter reading
Bluetooth	Low power usage	Short range; Low data rate; Weak security; Prone to interference from IEEE 802.11 WLAN	HAN	Home automation
Mobile Broadband Wireless Access	High bandwidth; High mobility; Low latency	Moderate data rate; Communication infrastructure is not readily available; High cost	NAN; WAN	Broadband communication for plug-in electric vehicles, Wireless backhaul for electric grid monitoring and SCADA systems
Digital Microwave Technology	High data rate; Long range (up to 60 km)	Susceptible to precipitation; Multi-path interference; Additional latency due to encryption for security	HAN; NAN	Alarms between DERs and distribution substation feeder
Wireless LAN	Robust; High speed point-to-point and point-to-multipoint communication; Minimum interference to out-of-band users; High data rate; Easy installation; Low cost	Slow data transmission due to electromagnetic interference in high voltage environment; Prone to RF interference; Limited availability of industrial-hardened wireless LAN equipment; Complex design for high reliability and availability	HAN; NAN; AMI	Distribution substation automation and protection, Monitoring and control of DERs
WiMAX	High data rate; Long range	Network speed degrades with increasing distance; High power in licensed spectrum; Costly RF hardware; High frequencies do not penetrate through obstacles; Licensing required for low frequencies	HAN; AMI	Wireless Automatic Meter Reading; Real-time pricing; Outage Detection and Restoration
Cellular Communications	Extensive data coverage; Improvement in QoS with recent growth in cellular technology	Range depends on the availability of cellular service; Delays due to call establishment; Large data exchange due to call drops; High monthly fees for individual connection; Expensive call costs	HAN; NAN; AMI	SCADA; Monitoring and metering of remote DERs

Powerline Communication	High speed; Low cost due to already available infra-structure; Low latency; Widespread availability	Harsh and noisy medium; Low band-width; Signal quality is affected by the network topology, the number and type of the devices connected to the power-lines, and wiring distance between transmitter and receiver; Sensitive to disturbances	HAN; NAN; AMI	Automatic Meter Reading; Low voltage distribution
Digital Sub-scriber Lines	High speed; Low cost due to already available infra-structure; Low latency; High Capacity; High data rate; Long range; Wide-spread availability	Throughput depends on the distance of the subscriber from the serving telephone exchange; Lack of standardization; High installation cost for new connections; Requires regular maintenance; Not fea-sible in low-density area due to the high cost of installing fixed infrastructure	HAN; NAN; WAN; AMI	Smart Grid city; Smart metering
Optical networks	High speed; Large band-width; High degree of reliability; Long range	High cost; Interoperability	NAN; WAN	Physical network infrastructure
IEEE 802.22 CR WRAN	High data rate; Long range; Immunity to inter-ference; Extensive data coverage; Adaptive power levels	Susceptible to disruptions die to PU activity; Security; Interoperability	HAN; NAN; WAN; AMI	Smart metering; SCADA, Demand Response Manage-ment; Wide Area Monitoring Control and Protection

APPENDIX B. Signal mapping tags and descriptions

Table 1. Signal mapping tags for measurements and corresponding descriptions for the movable unit, the signal names are the same for other REF615 with corresponding object names: Protection_JK-T.KK-TRB.REF615, Protection_JK-T.RK-TRB.REF615 or Protection_JK-T.KK-PV.REF615.

Tag	Description
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotW.instMag	total real power
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotW.t	total real power timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotW.q	total real power quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVA.instMag	total apparent power
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVA.t	total apparent power timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVA.q	total apparent power quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVAr.instMag	total reactive power
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVAr.t	total reactive power timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotVAr.q	total reactive power quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotPF.instMag	total power factor
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotPF.t	total power factor timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_TotPF.q	total power factor quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsA.cVal	phase A current
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsA.t	phase A current timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsA.q	phase A current quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsB.cVal	phase B current
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsB.t	phase B current timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsB.q	phase B current quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsC.cVal	phase C current
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsC.t	phase C current timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_A.phsC.q	phase C current quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsA.cVal	phase A phase voltage
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsA.t	phase A phase voltage timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsA.q	phase A phase voltage quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsB.cVal	phase B phase voltage
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsB.t	phase B phase voltage timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsB.q	phase B phase voltage quality indicator
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsC.cVal	phase C phase voltage
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsC.t	phase C phase voltage timestamp
VTT:Protection_JK-T.KK-TL3.REF615:M_EA_PhV.phsC.q	phase C phase voltage quality indicator

VTT:Protection_JK-T.KK-TL3.REF615:C_EA_Pos.stVal	circuit breaker position [10 = Close, 01 = Open]
VTT:Protection_JK-T.KK-TL3.REF615:C_EA_Pos.t	circuit breaker position timestamp

APPENDIX C. Program code for data transfer

```
using System;
using System.Threading;
using lib60870;
using StackExchange.Redis;

namespace testclient3
{
       class MainClass
       {
        private static void ConnectionHandler(object parameter, Connec-
tionEvent connectionEvent)
        {
            switch (connectionEvent)
            {
                case ConnectionEvent.OPENED:Console.WriteLine
                           ("Connected");
                    break;
                case ConnectionEvent.CLOSED:Console.WriteLine
                           ("Connection closed");
                    break;
                case ConnectionEvent.STARTDT_CON_RECEIVED:Console.
                           WriteLine ("STARTDT CON received");
                    break;
                case ConnectionEvent.STOPDT_CON_RECEIVED:Console.
                           WriteLine("STOPDT CON received");
                    break;
            }
        }
        private static string qualityHandler(string q)
        {
            string redisQ;
            if (q == " Invalid=True]")
            {
                redisQ = "0";
            }
            else
            {
                redisQ = "1";
            }
            return redisQ;
```

```
}
private static bool asduReceivedHandler(object parameter, ASDU
      asdu, Connection con)
{
    Console.WriteLine(asdu.ToString());
    string controlValue2 = "100";
    string qualityRedis = "20";
    double timetage = 100;
    DateTime Jan11970 = new DateTime(1970, 1, 1, 0, 0, 0,
            DateTimeKind.Utc);
    ConnectionMultiplexer redisConn = ConnectionMultiplex
            er.Connect("localhost");
    IDatabase redDb = redisConn.GetDatabase();
    if (asdu.TypeId == TypeID.M_ME_TF_1)
    {
        for (int i = 0; i < asdu.NumberOfElements; i++)
        {
            var mfv = (MeasuredValueShortWithCP56Time2a)asdu.
                   GetElement(i);
            Console.WriteLine("  IOA: " + mfv.ObjectAddress + "
                   float value: " + mfv.Value);
            Console.WriteLine("   " + mfv.Quality.ToString());
            Console.WriteLine("   " + mfv.Timestamp.ToString());
            Console.WriteLine("   " + mfv.Timestamp. Get
                   DateTime() .ToString());
            Console.WriteLine("   " + DateTime.UtcNow);
            if (mfv.ObjectAddress == 701)
            {
                redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
                   REF615:M_EA_TotW.instMag", mfv.Value);
                timetage = (mfv.Timestamp.GetDateTime() -
                   Jan11970).TotalMilliseconds;
                redDb. StringSet("VTT:Protection_JK-T.KK-TL3.
                   REF615:M_EA_TotW.t", timetage);
                qualityRedis = qualityHandler (mfv.Quality.
                   ToString().Split(',')[4]);
                redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
                   REF615:M_EA_TotW.q", qualityRedis);
                Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
                   REF615:M_EA_TotW updated to Redis");
            }
            else if (mfv.ObjectAddress == 703)
            {
```

```
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVA.instMag", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVA.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVA.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVA updated to Redis");
}
else if (mfv.ObjectAddress == 702)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVAr.instMag", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVAr.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVAr.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotVAr updated to Redis");
}
else if (mfv.ObjectAddress == 704)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotPF.instMag", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotPF.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotPF.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_TotPF updated to Redis");
}
```

```
else if (mfv.ObjectAddress == 711)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsA.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsA.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsA.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsA updated to Redis");
}
else if (mfv.ObjectAddress == 712)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsB.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsB.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsB.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsB updated to Redis");
}
else if (mfv.ObjectAddress == 713)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsC.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsC.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsC.q", qualityRedis);
```

```
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_A.phsC updated to Redis");
}
else if (mfv.ObjectAddress == 721)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsA.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsA.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsA.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsA updated to Redis");
}
else if (mfv.ObjectAddress == 722)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsB.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsB.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsB.q", qualityRedis);
    Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsB updated to Redis");
}
else if (mfv.ObjectAddress == 723)
{
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsC.cVal", mfv.Value);
    timetage = (mfv.Timestamp.GetDateTime() -
      Jan11970).TotalMilliseconds;
    redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
      REF615:M_EA_PhV.phsC.t", timetage);
    qualityRedis = qualityHandler(mfv.Quality.
      ToString().Split(',')[4]);
```

```
            redDb.StringSet("VTT:Protection_JK-T.KK-TL3.
              REF615:M_EA_PhV.phsC.q", qualityRedis);
            Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
              REF615:M_EA_PhV.phsC updated to Redis");
        }
        else
        {
            Console.WriteLine("Unknown indication ad
              dress!");
        }
    }
}
else
{
    Console.WriteLine("Unknown message type!");
}
string cValue = redDb.StringGet("VTT:Protection_JK-T.KK-TL3.
 REF615:C_EA_Pos.stVal");
Console.WriteLine("VTT:Protection_JK-T.KK-TL3.
 REF615:C_EA_Pos.stVal received");
if (cValue != "nil" && cValue != controlValue2)
{
    if (cValue == "10")
    {
      con.SendControlCommand(CauseOfTransmission.ACTIVATION,
        101, new SingleCommand(100, true, false, 0));
        Console.WriteLine("control signal OPEN sent!");
    }
    else if (cValue == "01")
    {
      con.SendControlCommand(CauseOfTransmission.ACTIVATION,
        101, new SingleCommand(100, true, false, 0));
        Console.WriteLine("control signal CLOSED sent!");
    }
    else
    {
        Console.WriteLine("Control signal is nil/unknown!");
    }
}
else if (cValue == controlValue2)
{
    Console.WriteLine("No change in the control signal!");
}
```

```
            else
            {
                Console.WriteLine("Unknown control signal!");
            }
            controlValue2 = cValue;
            return true;
        }

        public static void Main (string[] args)
             {
            Console.WriteLine("Using lib60870.NET version " + Library
             Common.GetLibraryVersionString());
            Connection con = new Connection("10.0.0.11", 2404);
            con.DebugOutput = true;
            con.SetASDUReceivedHandler(asduReceivedHandler, null);
            con.SetConnectionHandler(ConnectionHandler, null);
            bool running = true;
            Console.CancelKeyPress += delegate (object sender, Console
             CancelEventArgs e)
            {
                e.Cancel = true;
                running = false;
            };
            con.Connect();  //connecting to  COM600 slave client
            Thread.Sleep(5000);
            con.SendTestCommand(1);
            while (running)
            {
                 con.SendInterrogationCommand(CauseOfTransmission.
                ACTIVATION, 1, QualifierOfInterrogation.STATION);
                Thread.Sleep(5000);
            }
            Console.WriteLine("CLOSE");
            con.Close();
            Console.WriteLine("Press any key to terminate...");
            Console.ReadKey();
        }
    }
}
```

APPENDIX D. Delay testing results

Table 3. Delay testing results of measurement data transfer from VTT's local Redis RTDB to the cloud-based Redis RTDB.

Timestamp 1 (UTC)	Timestamp 2 (UTC)	Duration
21.11.2017 13:29:52.897	21.11.2017 13:29:53.104	0:00:00.207
21.11.2017 13:29:54.545	21.11.2017 13:29:54.838	0:00:00.293
21.11.2017 13:29:56.295	21.11.2017 13:29:56.588	0:00:00.293
21.11.2017 13:29:57.961	21.11.2017 13:29:58.337	0:00:00.376
21.11.2017 13:29:59.862	21.11.2017 13:30:00.088	0:00:00.226
21.11.2017 13:30:01.478	21.11.2017 13:30:01.850	0:00:00.372
21.11.2017 13:30:03.088	21.11.2017 13:30:03.355	0:00:00.267
21.11.2017 13:30:04.691	21.11.2017 13:30:05.092	0:00:00.401
21.11.2017 13:30:06.313	21.11.2017 13:30:06.589	0:00:00.276
21.11.2017 13:30:07.906	21.11.2017 13:30:08.090	0:00:00.184
21.11.2017 13:32:32.385	21.11.2017 13:32:32.589	0:00:00.204
21.11.2017 13:32:33.984	21.11.2017 13:32:34.338	0:00:00.354
21.11.2017 13:32:35.640	21.11.2017 13:32:35.843	0:00:00.203
21.11.2017 13:32:37.213	21.11.2017 13:32:37.595	0:00:00.382
21.11.2017 13:32:38.838	21.11.2017 13:32:39.088	0:00:00.250
21.11.2017 13:32:40.445	21.11.2017 13:32:40.838	0:00:00.393
21.11.2017 13:32:42.091	21.11.2017 13:32:42.346	0:00:00.255
21.11.2017 13:32:43.797	21.11.2017 13:32:44.092	0:00:00.295
21.11.2017 13:32:45.459	21.11.2017 13:32:45.844	0:00:00.285
21.11.2017 13:32:47.046	21.11.2017 13:32:47.338	0:00:00.292
21.11.2017 13:38:41.232	21.11.2017 13:38:41.832	0:00:00.600
21.11.2017 13:38:42.864	21.11.2017 13:38:43.087	0:00:00.223
21.11.2017 13:38:44.518	21.11.2017 13:38:44.852	0:00:00.334
21.11.2017 13:38:46.080	21.11.2017 13:38:46.339	0:00:00.259
21.11.2017 13:38:47.683	21.11.2017 13:38:48.090	0:00:00.407
21.11.2017 13:38:49.427	21.11.2017 13:38:49.856	0:00:00.429
21.11.2017 13:38:51.037	21.11.2017 13:38:51.349	0:00:00.312
21.11.2017 13:38:52.693	21.11.2017 13:38:53.090	0:00:00.397
21.11.2017 13:38:54.304	21.11.2017 13:38:54.591	0:00:00.287
21.11.2017 13:38:55.895	21.11.2017 13:38:56.088	0:00:00.193

Table 4. Delay testing result of control signal data transfer from RSE’s local Redis RTDB to VTT’s local Redis RTDB.

Timestamp 1 (UTC)	Timestamp 2 (UTC)	Duration
21.11.2017 13:59:37	21.11.2017 14:01:06	0:01:28

Table 5. Delay testing results of data transfer from ABB REF615 relay to VTT’s local Redis.

Timestamp 1 (UTC)	Timestamp 2 (UTC)	Duration
21.11.2017 10:06:27	21.11.2017 10:07:00	0:00:33
21.11.2017 10:12:49	21.11.2017 10:12:56	0:00:07
21.11.2017 11:56:13	21.11.2017 11:56:20	0:00:07