



---

Interoperable solutions for implementing holistic **FLEX**ibility  
services in the distribution **GRID**

---

**Publishable report on FLEXiGRID interoperability  
environment**

**Deliverable 5.8**

**WP5**

**Grant agreement: 864579**  
From 1<sup>st</sup> October 2019 to 30<sup>th</sup> September 2023

**Prepared by: ATOS**

**Date: 30/09/2022**



This project has received funding from the European Union's Horizon 2020 research and innovation programme under service agreement No 864579

**Disclaimer:** The sole responsibility for any error or omissions lies with the editor. The content does not necessarily reflect the opinion of the European Commission. The European Commission is also not responsible for any use that may be made of the information contained herein

## DELIVERABLE FACTSHEET

Deliverable no.	D5.8
Responsible Partner	ATOS
WP no. and title	WP5 Cyber ICT layer and Interoperability environment
Task no. and title	Task 5.4 FLEXIGRID middleware and platform adoption Task 5.5 End-user interface development
Version	0.8
Version Date	30/09/2022

Dissemination level	
X	PU → Public
	PP → Restricted to other programme participants (including the EC)
	RE → Restricted to a group specified by the consortium (including the EC)
	CO → Confidential, only for members of the consortium (including the EC)

## Approvals

	Company
Author/s	ATOS, CIRCE, EDYNA, HEP-ODS, HYPERTECH, IOSA, LINKS, UNICAN, VERD
Task Leader	ATOS
WP Leader	ATOS

## Document History

Revision	Date	Main Modification	Author
0.1	12/08/2022	ToC	A. Medela (ATOS)
0.2	26/08/2022	Executive Summary, Sections 1 & 2	A. Medela (ATOS)
0.3	02/09/2022	Initial content in Sections 6 & 7	A. Medela, D. R. Salinas (ATOS)
0.4	06/09/2022	Additional content in Section 6	A. Medela (ATOS)
0.5	12/09/2022	Updated document structure	A. Medela, S. Palacios (ATOS)
0.6	15/09/2022	Additional content in Executive Summary, Abbreviations, Sections 1, 2 & 7	A. Medela, S. Palacios (ATOS)
0.7	27/09/2022	Internal Review	T. Antic, M. Miletić (UNIZG-FER), A. Lostale (CIRCE)
0.8	30/09/2022	Updates after the internal review	A. Medela, S. Palacios (ATOS)

## ABBREVIATIONS

<b>AMI:</b>	Advanced Metering Infrastructure
<b>AMQP:</b>	Advanced Message Queuing Protocol
<b>API:</b>	Application Programming Interface
<b>ASN:</b>	Autonomous System Number
<b>BMCD:</b>	Building Monitoring and Control Dispatch
<b>CA:</b>	Consortium Agreement
<b>CC:</b>	Communication Committee
<b>CIM:</b>	Common Information Model
<b>COSEM:</b>	Companion Specification for Energy Metering
<b>CP:</b>	Comfort Profiling
<b>CURL:</b>	Client URL
<b>DER:</b>	Distributed Energy Resources
<b>DHW:</b>	Domestic Hot Water
<b>DLMS:</b>	Device Language Message Specification
<b>DLP:</b>	Data Loss Prevention
<b>DMP:</b>	Data Management Plan
<b>DMS:</b>	Distribution Management System
<b>DNP:</b>	Distributed Network Protocol
<b>DoA:</b>	Description of Action
<b>DR:</b>	Demand Response
<b>DSO:</b>	Distribution System Operators
<b>EC:</b>	European Commission
<b>EMS:</b>	Energy Management System
<b>ESCo:</b>	Energy Service Company
<b>ETL:</b>	Extract, Transform, Load
<b>ETSI:</b>	European Telecommunications Standards Institute
<b>EV:</b>	Electric Vehicle
<b>FPI:</b>	Fault Passage Indicator
<b>FTP:</b>	File Transfer Protocol
<b>FTPS:</b>	FTP-SSL or FTP-secure
<b>FUSE:</b>	Framework for Utilities and Services
<b>GA:</b>	Grant Agreement
<b>GA:</b>	General Assembly
<b>GOOSE:</b>	Generic Object-Oriented substation event
<b>GPRS:</b>	General Packet Radio Services
<b>GSSE:</b>	Generic Substation Status Event
<b>GUI:</b>	Graphical User Interface
<b>HAN:</b>	Home Area Network
<b>HMI:</b>	Human-Machine Interfaces
<b>HTTP:</b>	HyperText Transfer Protocol
<b>HTTPS:</b>	HyperText Transfer Protocol Secure
<b>HV:</b>	High Voltage

<b>ICT:</b>	Information and Communications Technology
<b>IDS:</b>	Intrusion Detection System
<b>IEC:</b>	International Electrotechnical Commission
<b>IED:</b>	Intelligent Electronic Device
<b>IEEE:</b>	Institute of Electrical and Electronics Engineers
<b>IoT:</b>	Internet of Things
<b>IP:</b>	Internet Protocol
<b>IPR:</b>	Intellectual Property Right
<b>ISO:</b>	International Organization for Standardization
<b>IT:</b>	Information Technology
<b>JWT:</b>	JSON Web Token
<b>KPI:</b>	Key Performance Indicator
<b>LAN:</b>	Local Area Network
<b>LV:</b>	Low Voltage
<b>M2M:</b>	Machine to Machine
<b>MITM:</b>	Man In The Middle
<b>MMS:</b>	Manufacturing Message Specification
<b>MQTT:</b>	Message Queuing Telemetry Transport
<b>MV:</b>	Medium Voltage
<b>NAN:</b>	Near-me Area Network
<b>NGSI:</b>	Next Generation Service Interface
<b>NIST:</b>	National Institute of Standards and Technology
<b>OAS:</b>	OpenAPI Specification
<b>OASIS:</b>	Organization of the Advancement of Structure Information Standards
<b>OPC:</b>	Open Platform Communications
<b>OPC:</b>	OLE for Process Control
<b>OpenADR:</b>	Open Automated Demand Response
<b>OLE:</b>	Object Linking and Embedding
<b>OSI:</b>	Open System Interconnection
<b>P2H:</b>	Power to Hydrogen
<b>PC:</b>	Project Coordinator
<b>PH:</b>	Project Handbook
<b>PHEV:</b>	Plug-in Hybrid Electric Vehicle
<b>PII:</b>	Personal Identifiable Information
<b>PKCE:</b>	Proof Key for Code Exchange
<b>PLC:</b>	Power-Line Communication
<b>PMU:</b>	Phasor Measuring Unit
<b>PPP:</b>	Point-to-Point Protocol
<b>PV:</b>	PhotoVoltaic
<b>QUIC:</b>	Quick UDP Internet Connections
<b>R&amp;D:</b>	Research and Development
<b>RES:</b>	Renewable Energy Source
<b>REST:</b>	Representational State Transfer

<b>RFC:</b>	Request for Comments
<b>RTU:</b>	Remote Terminal Unit
<b>SAREF:</b>	Smart Appliances REference
<b>SCADA:</b>	Supervisory Control And Data Acquisition
<b>SFTP:</b>	Secure File Transfer Protocol or SSH File Transfer Protocol
<b>SGAM:</b>	Smart Grids Architecture Model
<b>SNTP:</b>	Simple Network Time Protocol
<b>SS:</b>	Secondary Substation
<b>TCP:</b>	Transmission Control Protocol
<b>TDR:</b>	Time-Domain Reflectometer
<b>TESFP:</b>	Thermal Energy Storage and Flexibility Profiling
<b>TLS:</b>	Transport Layer Security
<b>TMC:</b>	Thermal Model Calculation
<b>TSO:</b>	Transmission System Operator
<b>UC:</b>	Use Case
<b>UI:</b>	User Interface
<b>UML:</b>	Unified Modelling Language
<b>URL:</b>	Uniform Resource Locator
<b>VES:</b>	Virtual Energy Storage
<b>VPN:</b>	Virtual Private Network
<b>VTES:</b>	Virtual Thermal Energy Storage
<b>WAN:</b>	Wide Area Network
<b>WP:</b>	Work Package

## DISCLAIMER OF WARRANTIES

*"This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No 864579".*

This document has been prepared by FLEXIGRID project partners as an account of work carried out within the framework of the EC-GA contract no 864579.

Neither Project Coordinator, nor any signatory party of FLEXIGRID Project Consortium Agreement, nor any person acting on behalf of any of them:

- (a) makes any warranty or representation whatsoever, express or implied,
  1. with respect to the use of any information, apparatus, method, process, or similar item disclosed in this document, including merchantability and fitness for a particular purpose, or
  2. that such use does not infringe on or interfere with privately owned rights, including any party's intellectual property, or
  3. that this document is suitable to any particular user's circumstance; or
- (b) assumes responsibility for any damages or other liability whatsoever (including any consequential damages, even if Project Coordinator or any representative of a signatory party of the FLEXIGRID Project Consortium Agreement, has been advised of the possibility of such damages) resulting from your selection or use of this document or any information, apparatus, method, process, or similar item disclosed in this document.

## EXECUTIVE SUMMARY

FLEXIGRID project comprises four pilot sites that offer a significant amount of data from diverse sources for a set of various services to benefit from them. There is an interest to make ends meet and provide the means to facilitate the required interactions and provide a mechanism to make it easier for end-users to consult and interpret the data produced by the system.

The work described in this report (D5.8) takes place in the scope of WP5 “Cyber ICT layer and Interoperability environment” (where ICT stands for Information and Communications Technology), which comprises five different tasks, led by technical partners and covering key aspects to make sure FLEXIGRID platform satisfies requirements to be truly interoperable. The main objectives of these tasks are:

- T5.1 “FLEXIGRID ICT architecture definition” includes the collection and documentation of the final organisation of the components of the project demo sites, illustrating the interconnected grid segments and elements, both the assets of the electrical grid and the electronic equipment to work on an integrated architecture seeking interoperability of the developments and further exploitation potential of the applications.
- T5.2 “Protocols and standards, interoperability and FLEXIGRID CIM” analyses the existing standards that affect the implementation of the project, having in mind also a wider view of the elements of the energy value chain covered. This will include the analysis and selection of the specific communication standards or ontologies suitable for the project demo. As a second step after the identification of the standards, the technical partners in the consortium will define a Common Information Model (CIM) based on all the data that is going to be managed and the different sources that generate it.
- T5.3 “Cybersecurity requirements, Access Control and Data Privacy mechanisms” performs an exhaustive analysis of the threats and risks derived from the procedures to deliver information and commands to the equipment participating in the project. The areas covered in the task to perform the analysis are: 1) Equipment security; 2) Communication security; 3) Data security; 4) Platform integration security.
- T5.4 “FLEXIGRID middleware and platform adoption” aims at integrating the different software modules developed during the project. To do so, FUSE platform will act as the glue for the integration of the different components and platforms coming from the Distribution System Operators (DSOs) and manufacturers participating in the project, including the development of the specific adapters shown in the lower levels of the Information Technology (IT) architecture of the project concept according to the results of Tasks 5.1 and 5.2.
- T5.5 “End-user interface development” focuses on the front end of the FLEXIGRID platform, which be tailored to the specifics required by each one of the Pilot Cases. Specifically, a common web-based interface shall be designed and accompanied by developed visual analytics methods and schemes for making available the most crucial information for the grid manager and its associated partners.

The result of all these tasks were different communication paths and implementations which are collected in this document as a summary of the technical approaches and the adaptation

based on the different demonstration site needs. The latter is found in sections 2 to 6, and the final comments and lessons learned by each developer are summarised in section 7.

All in all, the work depicted in this report is the publishable summary of the activities in WP5 which in turn were initially collected in project deliverables D5.1 "FLEXIGRID ICT Platform architecture – M12" [1], D5.2 "FLEXIGRID ICT Platform architecture – M24" [2], D5.3 "Protocols and standards definition" [3], D5.4 "FLEXIGRID Common Information Model (CIM)" [4], D5.5 "Platform cybersecurity mechanisms" [5], D5.6 "FLEXIGRID ICT Platform" [6] and D5.7 "Web-based end users' interfaces" [7].

## TABLE OF CONTENTS

1.	INTRODUCTION .....	14
1.1.	Scope and objectives.....	14
1.2.	Links with other tasks.....	14
1.3.	Structure of the document.....	15
2.	T5.1 FLEXIGRID ICT architecture definition .....	16
2.1.	Methodology.....	16
2.2.	FLEXIGRID's Reference Architecture. Logical View .....	17
	Reference Logical Architecture .....	17
2.3.	FLEXIGRID's Reference Architecture. Deployment View .....	19
2.4.	FLEXIGRID's Reference Architecture. Implementation View .....	20
	Grid infrastructure.....	20
	FUSE.....	21
	Applications.....	23
2.5.	FLEXIGRID's Reference Architecture. Process View .....	27
2.6.	FLEXIGRID'S Reference Architecture. Scenarios View.....	27
3.	T5.2 Protocols and standards, interoperability and FLEXIGRID CIM.....	28
3.1.	Protocols.....	28
	Protocols used in FLEXIGRID for data collection in the field.....	28
	Protocols used in FLEXIGRID for data sharing.....	34
3.2.	Data Models .....	36
	Common Information Model (CIM) .....	36
	FIWARE .....	37
	OpenADR.....	37
	SAREF.....	38
3.3.	Protocols and standards.....	38
	Representation Standards and FLEXIGRID Project.....	38
	FLEXIGRID Common Information Model design guidelines .....	40
3.4.	FLEXIGRID Common Information Model.....	41
	Methodology to define entities required for the FLEXIGRID CIM.....	41
4.	T5.3 Cybersecurity requirements, access control and data privacy mechanisms .....	47
4.1.	Cyber-security in smart grid: State of the Art .....	50
4.2.	Risks and threats analysis.....	55
	Impact on costs .....	59

4.3. Cyber-security framework design .....	61
Architecture security design .....	61
Pilot security .....	68
4.4. Secure platform implementation details .....	71
5. T5.4 Interoperability in FLEXIGRID's ICT platform .....	73
5.1. Interoperability in the Spanish demonstrator .....	73
OPC server in the Spanish Demo Site .....	74
5.2. Interoperability in the Greek demonstrator .....	75
5.3. Interoperability in the Croatian demonstrator .....	77
5.4. Interoperability in the Italian demonstrator .....	79
6. T5.5 FLEXIGRID's web-based end user interfaces .....	80
6.1. Graphical user interfaces implementation .....	80
What is Kibana and how it works in FLEXIGRID .....	80
What is Dash Plotly and how it works in FLEXIGRID .....	81
6.2. Data visualization for end users .....	81
7. CONCLUSIONS .....	83
8. REFERENCES .....	87
9. ANNEX 1 – FLEXIGRID logical architecture per demonstrator .....	92
9.1. Spanish Demonstrator .....	92
Scenario 1 .....	92
Scenarios 2 and 3 .....	93
9.2. Greek Demonstrator .....	94
9.3. Croatian Demonstrator .....	94
9.4. Italian Demonstrator .....	97
10. ANNEX 2 – FLEXIGRID deployment view per demonstrator .....	99
10.1. Spanish Demonstrator .....	99
Scenario 1 .....	99
Scenarios 2 and 3 .....	100
10.2. Greek Demonstrator .....	101
10.3. Croatian Demonstrator .....	102
10.4. Italian Demonstrator .....	104
11. ANNEX 3 – FLEXIGRID process view .....	106
11.1. Demonstrators and Sequence Diagrams .....	106
Spanish Demonstrator .....	107
Greek Demonstrator .....	114

Croatian Demonstrator .....	118
Italian Demonstrator .....	121
12. ANNEX 4 – FLEXIGRID scenarios view .....	124
12.1. Functional Requirements .....	124
12.2. Non-Functional Requirements .....	134
13. ANNEX 5 – FLEXIGRID CIM entities creation example .....	139
14. ANNEX 6 - Cybersecurity in the 4 areas of interest in a smart grid .....	143
14.1. Equipment Security .....	143
14.2. Communication Security .....	144
14.3. Data Security .....	144
14.4. Platform Integration Security.....	145

## LIST OF FIGURES

Figure 1. Relationships among D5.8 and FLEXIGRID tasks .....	15
Figure 2. Kruchten's 4+1 view model of software architecture. [9] .....	16
Figure 3. FLEXIGRID's reference logical architecture. ....	17
Figure 4. FLEXIGRID's Energy Box architecture .....	20
Figure 5. Grid equipment logical diagram .....	21
Figure 6. Single adaptor logical diagram .....	22
Figure 7. Common Information Model logical diagram .....	22
Figure 8. Unified API logical diagram .....	23
Figure 9. Forecast and grid operation component diagram .....	24
Figure 10. Grid congestion management component diagram .....	24
Figure 11. Building Monitoring and Control Dispatch Module Component diagram .....	25
Figure 12. Thermal Energy Storage and Flexibility Profiling Module diagram .....	26
Figure 13. Comfort Profiling Module diagram .....	26
Figure 14. Thermal Model Calculation Module diagram .....	26
Figure 15. IEC 101 and IEC 104 protocol stacks (OSI Model). Image based on the information available in [15] .....	29
Figure 16. IEC 61850 protocol stack [15] .....	30
Figure 17. DLMS possible uses and implementations [21] .....	31
Figure 18. DLMS/COSEM protocol stack [23] .....	32
Figure 19. Modbus communication stack [25] .....	32
Figure 20. Z-wave protocol stack [28] .....	34
Figure 21. OPC UA protocol stack [15] .....	36
Figure 22. Main elements in the NGSI data model .....	39
Figure 23. Methodology to define the entities of the FLEXIGRID CIM .....	41
Figure 24. Common grid entities .....	43
Figure 25. Preliminary list of FLEXIGRID CIM entities .....	44
Figure 26. Screenshot of a grid topology modelling software .....	45
Figure 27. IEC 61970-301 CIM standard classes for representing the electrical view of the distribution grid .....	45
Figure 28. List of objects generated by DigSilent for the Spanish demo using CIM standard ....	46
Figure 29. Key elements in Smart Grids .....	48
Figure 30. EnergyShield levels and dimensions .....	51
Figure 31. EnergyShield Asset dimension and domains .....	52
Figure 32. EnergyShield Continuity dimension and domains .....	52
Figure 33. EnergyShield Access and Trust dimension and domains .....	53
Figure 34. EnergyShield Operation dimension and domains .....	54
Figure 35. EnergyShield Defence dimension and domains .....	54
Figure 36. EnergyShield Security Governance dimension and domains .....	55
Figure 37.- Most significant cyber-attacks from 2006 to 2020 .....	60
Figure 38. Communications in the Spanish demo site .....	73
Figure 39. Sample of the OPC GUI showing loads measurements in several nodes .....	75
Figure 40. Communication in the Greek demo site .....	75
Figure 41. Sample of GET query performed on the Unified API .....	76
Figure 42. Communications in the Croatian demo site .....	77

Figure 43. Croatian demo site: RESTFUL APIs for flexibility services .....	77
Figure 44. GET request via Postman in the Croatian demo site .....	78
Figure 45. Croatian demo site: AMQP protocol for event-based data .....	78
Figure 46. Communications in the Italian demo site .....	79
Figure 47. Sample of Kibana Dashboard for Greek Pilot .....	81
Figure 48. Sample of Dash Dashboard for the Greek Pilot .....	82
Figure 49. Sample of Dash Dashboard for the Croatian Pilot .....	82
Figure 50. Diagram of software modules used in Spanish demonstrator's scenario 1.....	93
Figure 51. Diagram of software modules used in Spanish demonstrator's scenarios 2 and 3. ..	93
Figure 52. Diagram of software modules used in the Greek demonstrator. ....	94
Figure 53. Diagram of software modules used in the Croatian demonstrator's distribution network flexibility and protection schemes coordination. ....	95
Figure 54. Diagram of software modules used in the Croatian demonstrator's thermal energy storage optimisation. ....	96
Figure 55. Diagram of software modules used in the Italian demonstrator.....	98
Figure 56. Deployment view of scenario 1 in the Spanish demonstrator.....	100
Figure 57. Deployment view of ORMAZABAL's substation of the future. ....	100
Figure 58. Deployment view of scenarios 2 and 3 in the Spanish demonstrator. ....	101
Figure 59. Deployment view of the Greek demonstrator. ....	102
Figure 60. Deployment view of the substations taking part in the Croatian demonstrator.....	103
Figure 61. Deployment view of the substations taking part in the Croatian demonstrator.....	103
Figure 62. Deployment view of the Italian demonstrator. ....	105
Figure 63. Spanish Use Case 1 Sequence Diagram (v0).....	112
Figure 64. Spanish Use Case 2 Sequence Diagram.....	113
Figure 65. Greek Use Cases 3 & 4 Sequence Diagram .....	117
Figure 66. Croatian Use Case Sequence Diagram .....	120
Figure 67. Italian Use Case Sequence Diagram .....	123
Figure 68. End-to-End Smart Grid communications high-level architecture .....	144
Figure 69. FUSE OpenAPI descriptions.....	146

## LIST OF TABLES

Table 1: Comparison between some protocols used in system automation in electrical substations [20].....	30
Table 2: Classification of Smart Grid Cyber-Attacks according to The CIA triad .....	57
Table 3 (Code): Example of setpoints in Greek demo site .....	76
Table 4: Use Cases and trials of Spanish Demonstrator.....	107
Table 5: Scenarios of Greek Demonstrator .....	114
Table 6: Scenarios of Croatian Demonstrator.....	118
Table 7: Scenarios of Italian Demonstrator.....	121
Table 8 (Code): Example of PowerTransformer entity.....	140

# 1. INTRODUCTION

This deliverable presents the steps and actions carried out in order to put in place the proper platform and graphic tools to be employed by FLEXIGRID to guarantee the system's interoperability and the easiness of use by the final users. This section introduces the scope and objective, as well as its relationship with other tasks of the project.

## 1.1. Scope and objectives

This deliverable ("*Publishable report on FLEXIGRID interoperability environment*") is the main outcome of the activities performed within the tasks that shape WP5.

Hence, the work done here is the continuation of the one carried out in the previous months in other WP5 tasks, reflected accordingly in D5.1 "*FLEXIGRID ICT architecture definition – Month 12*" [1], D5.2 "*FLEXIGRID ICT architecture definition – Month 24*" [2], D5.3 "*Protocols and standards definition*" [3], D5.4 "*FLEXIGRID Common Information Model (CIM)*" [4], D5.5 "*Platform cybersecurity mechanisms*" [5], D5.6 "*FLEXIGRID ICT platform*" [6] and D5.7 "*Web-based end users' interfaces*" [7], and recaps the steps taken in the achievement of the interoperability pursued within FLEXIGRID, as well as the preparation of a tool that permits seamless interactions of diverse end users.

All in all, these results will represent the main output coming out of WP5 and in the long term feed the diverse FLEXIGRID pilots.

## 1.2. Links with other tasks

Figure 1 below shows an approach to the relations this deliverable and its associated Tasks (namely, T5.1, T5.2, T5.3, T5.4 and T5.5) establishes with other WPs and activities within the project organization of work. It is important to note the picture focuses solely on D5.8, so not all relations between the rest of the WPs and tasks are presented.

More specifically, the activities performed in WP5 relate to other technical work packages, such as WP4 "*Development of Software services and modules*" or WP6 "*Demonstration campaigns*". In the end, the pilots will be the ones to make use of the platform and graphical interfaces proposed in WP5 to complement the integration activities and put into place various real-life use cases.

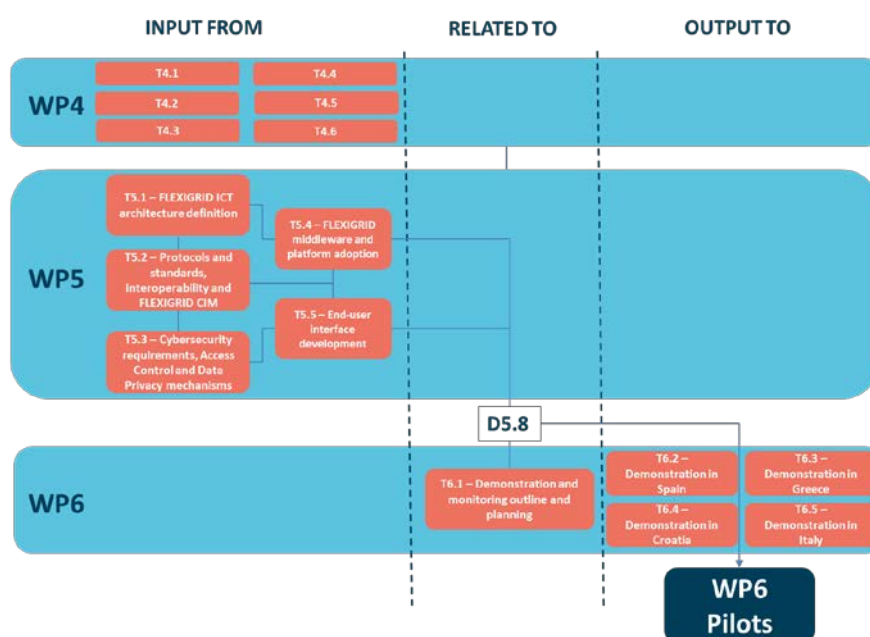


Figure 1. Relationships among D.5.8 and FLEXIGRID tasks

### 1.3. Structure of the document

This report is structured as follows:

**Section 2** provides a recap on the activities associated to T5.1 and its FLEXIGRID ICT architecture definition, initially introduced in Deliverable 5.1 [1] and further evolved in Deliverable 5.2 [2].

**Section 3** digs deep into the different protocols and standards evaluated and involved in FLEXIGRID, as well as the process followed to create the project's CIM, as reflected originally in Deliverable 5.3 [3] and Deliverable 5.4 [4] respectively.

**Section 4** evaluates cybersecurity in smart grids as of today, performs an analysis on risks and threats and proceeds to design FLEXIGRID's cybersecurity framework, as well as to define a series of details to implement a secure platform, topics presented in Deliverable 5.5 [5].

**Section 5** digs deep into the different solutions FLEXIGRID proposes to assure a proper interoperability takes place and sets up the platform for the immediate future, as discussed in Deliverable 5.6 [6].

**Section 6** introduces the graphical user interfaces (GUIs) which constitute the unified way to check relevant data and its associated key performance indicators values, along with interesting and easy to understand graphs, as reflected in Deliverable 5.7 [7].

Finally, **section 7** compiles a series of conclusions that wrap up the work carried out in WP5 tasks and hints to the next steps to take. These next steps include the usage of these solutions to grant a proper interaction with services and end users (namely WP4 and WP6) because of these tasks' activities.

## 2. T5.1 FLEXIGRID ICT architecture definition

### 2.1. Methodology

International Standard ISO/IEC/IEEE 42010 indicates that due to the extraordinary growth of complexity in modern systems, the application of architecting concepts, principles and procedures becomes essential to manage such intricacies and to help understanding the significance and key properties of the system's behaviour, structure and development. This understanding successively affects the system's feasibility, utility and maintenance. Consequently, and in order to ease the description of architectures to promote collaboration and communication between stakeholders, architecture frameworks and description languages have been created to systematize the common practices and protocols of architecting within different contexts. [8]

Within the conceptual model of an architecture description defined by ISO/IEC/IEEE 42010, architecture viewpoints are defined as the elements that set up the conventions to construct, interpret and analyse architecture views in order to address concerns held by one or more stakeholders. Viewpoints are then specified as requirements for conformance with the standard.

- *Kruchten's 4+1 view model* [9]: An efficient and flexible way to describe the architecture of software-intensive systems. It consists of 4 views (Logical, Development, Process and Physical) and an additional view called Scenarios that complements the others by describing a few use cases (UCs) (hence considered as a "+1").

The chosen architectural viewpoints to describe FLEXIGRID's architecture is Kruchten's 4+1 view model. This decision was promoted by the ease to understand and apply these views and the flexibility that the architectural model offers. Figure 2 shows a diagram with the proposed views accompanied by their corresponding stakeholders and major concern.

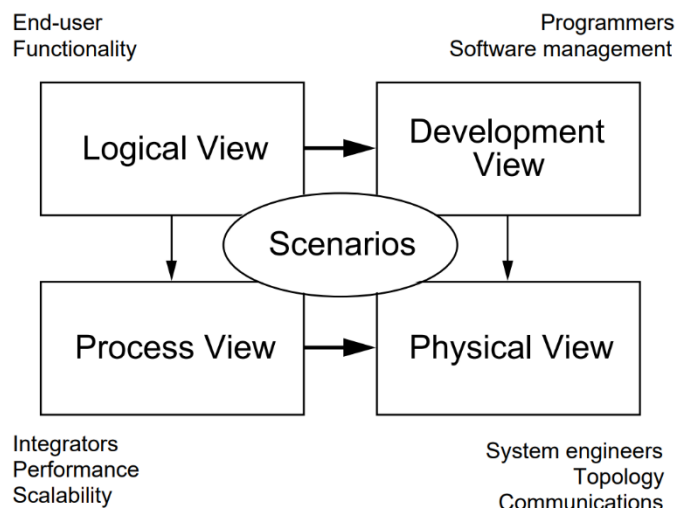


Figure 2. Kruchten's 4+1 view model of software architecture. [9]

The **logical view** represents the system's functional requirements. It is depicted in Deliverable 5.1 as a description of the static behaviour of the components that make up FLEXIGRID's architecture as a whole, and in each individual demo site.

Also present in that document, the physical view (or **deployment view**) represents the system's physical configuration of the software, i.e., how the executing software is mapped to processing nodes.

Both views, logical and deployment, are recapped in this report, since they will reflect the whole architectural approach of the FLEXIGRID platform and in this way, readers will find a go-to document where find it all compiled.

The development view, process view and scenarios view are a subject to address in this D5.2, which continues the work of D5.1. The development view, also called **implementation view**, should focus on the actual arrangement of software modules (in terms of libraries, or subsystems). The **process view** should describe the different processes (i.e., "tasks that form an executable unit" [9]) and how they interact with each other. Lastly, the **scenarios view** should focus on specific instances of generic use cases (a.k.a. scenarios) that will show how the 4 main views are joined together to exemplify and verify them.

## 2.2. FLEXIGRID's Reference Architecture. Logical View

### Reference Logical Architecture

A diagram representing FLEXIGRID's reference logical architecture can be found in Figure 3. The modules are separated in five layers corresponding to each of the SGAM (Smart Grids Architecture Model) interoperability layers [10]: Component, communication, information, function and business.

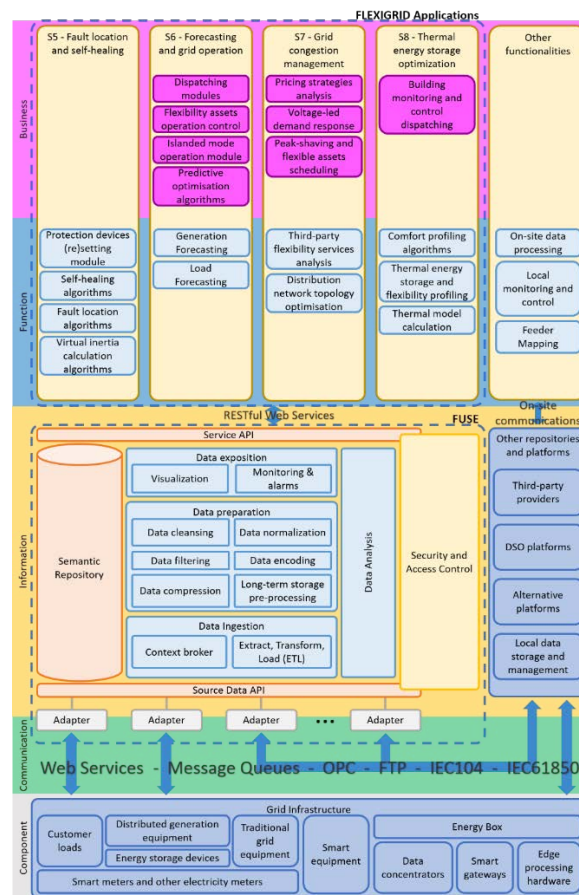


Figure 3. FLEXIGRID's reference logical architecture.

### *Component Layer*

Within the component layer, one can find the grid infrastructure available in FLEXIGRID's demo sites. It is composed of traditional and smart equipment, along with devices to retrofit and smarten conventional hardware.

The only component in the diagram with a non-self-explanatory name is the Energy Box, which represents one of FLEXIGRID's solution (S4). The reason of this is because it can perform various functionalities, such as data concentration, data gateway or edge processing (hence its location in the diagram, stretching out over these single-functionality components). For more information about the Energy Box, refer to D3.3 (Advanced Energy Box prototype).

### *Communication layer*

The different data flows between demo sites and FLEXIGRID's ICT platform are shown in the communication layer. Three data flows have been identified: from demo sites to FUSE; from demo sites to other platforms, such as a DSO's already existing platform; and from these platforms to FUSE (e.g., to get data from a third-party weather provider). Notice that all flows are bidirectional, representing the capability to send signals back to field devices. Moreover, there are many protocols available to support these communications. For additional details about the purpose of these protocols in a general perspective or for useful references, see D5.3 (Protocols and standards definition) [3].

### *Information layer*

As one of FLEXIGRID's solutions (S9), the information layer's main component is FUSE platform. A bottom-up analysis of its components shows that at the lower part of the information layer there are the adaptors, responsible for protocol adaptation and data harmonisation; and a source data API, to ultimately ingest data to the platform. Next components are the context broker and the ETL module. The former is in charge of handling real-time or near real-time data ingestion, while the latter manages batch data coming from buffers or historical databases. Following are data preparation modules that can be used to pre-process the data depending on the specific needs of the services that are ultimately exploiting it. These modules can be activated on-demand and include:

- Data filtering, such as noise removal or frequency filtering
- Data cleansing, for outlier correction, missing data filling, etc.
- Data normalization, for statistical normalization (standard score normalisation, feature scaling, etc.)
- Long-term storage pre-processing, for preparing data destined to be archived (e.g., format adaption or lossless compression algorithms)
- Data compression, such as dimensionality reduction, data source removal or down sampling
- Data encoding, such as categorical data encoding, binary-to-text, etc.

The top modules in FUSE platform are data exposition (for visualisation and monitoring) and an upper API for advances services to access the collected, harmonised data.

Additionally, extended along the aforementioned FUSE modules are: a semantic repository, a data analysis module, and a security and access control layer. The semantic repository is a database that also stores information about how the data is structured inside. Even though it provides other advantages (e.g., data validation), the main purposes of this type of database in

FUSE is to ease the development of data adaptors and data model mappings, fostering interoperability. More details about the data model information to be included in the semantic repository can be found in D5.4 (FLEXIGRID Common Information Model). The data analysis module is composed of a set of tools and frameworks available in FUSE for inspecting the data in order to discover trends and patterns that could give birth to different strategies for exploiting the data. Lastly, the security and access control layer encompass all the methodologies used and features available in FUSE for protecting the data and limiting who has access to it. Refer to D5.5 (Platform cybersecurity mechanisms) for more information about these mechanisms.

Parallel to FUSE, the information layer of the reference logical architecture also considers other repositories and platforms that are essential for demo site operations:

- Third party providers: To obtain external information such as weather data or energy tariffs
- DSO platforms: Already available at demo sites to support normal operation. The data that can flow from these platforms to FUSE include historical datasets or alternative data flows in which real-time data sharing is not possible for any reason
- Alternative platforms: These platforms are included in FLEXIGRID to support operations performed by a specific FLEXIGRID partner (e.g., VERD's platform or HYPERTECH's platform). There are multiple reasons for this parallelism: sensitive data management, extremely resource-demanding processes, particular business concerns, etc.
- Local data storage and management: This box comprises all hardware components available on-site that have capabilities for local data storage and management

#### *Function Layer and business layer*

Both the function layer and the business layer contain software elements closely related to FLEXIGRID's objectives (they are developed as part of FLEXIGRID's solutions). The modules in blue are considered as part of the functional layer for being more generic and for enabling smart grid use cases, and the modules in purple are considered as part of the business layer for supporting the creation of products and services. Moreover, these modules are grouped according to their contribution in FLEXIGRID's software solutions (S5-S8) and an additional group to sort those software functionalities that are expected to be executed locally or in FLEXIGRID's hardware solutions (S1-S4).

### 2.3. FLEXIGRID's Reference Architecture. Deployment View

As part of this activity, the details regarding the deployment view of each of FLEXIGRID's demonstration sites are described. To this end, block diagrams are used to illustrate the hardware equipment available in demo sites, indicating also the physical and virtual connections that they have with each other and with the data platforms available. For all these communications, the protocol is specified in black font, while the channel—or interface—is specified in blue. Unless otherwise specified, all the connections made with data platforms using HTTP or any other protocol over TCP are made via Internet, thus using the available communication channels for that purpose (router connected to the Internet infrastructure, or GPRS or a similar wireless technology).

For consistency, the colour code used for the blocks throughout this section follow this pattern: yellow for metering devices, green for gateways or other devices that communicate directly with

data platforms, blue for data platforms or any other software system, red for traditional energy devices, and grey for physical places. This exercise is available at Annex 2.

## 2.4. FLEXIGRID's Reference Architecture. Implementation View

This chapter incorporate the subtleties of every module displayed in the FLEXIGRID Logical View in Figure 3. The centre is set to portraying the functionality and data flow of every part, alongside its interactions with different modules when relevant.

### *Grid infrastructure*

This layer intends to satisfy two objectives. In the first place, the included devices will furnish the platform with real-time data (for example consumption, generation, emissions, power coefficient, thermal/electricity storage), obtained from smart or other edge devices. Also, this layer is capable to control the distribution of loads, hence controlling the micro grid. Their configuration will be partly dynamic and to some degree static, controlled by the manager/end-user of the facility.

### *Energy Box*

The Energy Box (Figure 4) has the role of a local data management system. Its use reduces the number of equipment communicating with the high layers of control and integrates them in one single device that holds several communication technologies, improving the efficiency control of the system. It is based on a multicore architecture with a non-blocking exchange structure that provides state-of-the-art capacities, offering not only domestic level benefits, but also other complex system requirements for most demanding environments and closer to a real time management. Therefore, the Energy Box is presented as an embedded and compact solution to monitor and manage intelligent devices in different kind of real scenarios.

In the FLEXIGRID demo sites, this component is going to collect the information from the field devices from the Greek and Spanish demo sites, adapting the field communication protocols to the one needed for the equipment. This information is sent to the FUSE platform for further treatment by the other modules of the project. The Energy Box also acts as a gateway receiving control signals from the FUSE platform and sending them to the field devices.



Figure 4. FLEXIGRID's Energy Box architecture

### *Smart Equipment*

Intelligent equipment: installed in the Secondary Substation (SS) of the future, it can, by means of remote orders from equipment in its communications network, carry out actions that give the electrical grid a flexible operation. It also allows autonomous operation based on the measurements (voltage and current, power, energies, angles...) and signals (alarms, activations, etc ...) obtained from the network of the different elements of the substation, in addition to using the information received by communications from other CT elements allowing a more

efficient and safe operation. For this reason, the Smart transformer and the LVB- Adibbo are designed with the capacity to use the existing communications and interconnections in the SS. Distributed generation equipment.

#### Traditional grid equipment

The development of "traditional" equipment for the electrical grid is made up of electro-mechanical equipment with limited functionality and little or no communication capacity both internally and externally with superior systems, has been a response for the deployment of the network. Transformers, LVB, cubicles, switches, etc ... are elements installed in the network with an autonomous operation of protection and effective control but limited in flexibility and evolution in the face of changes in the use of the network.

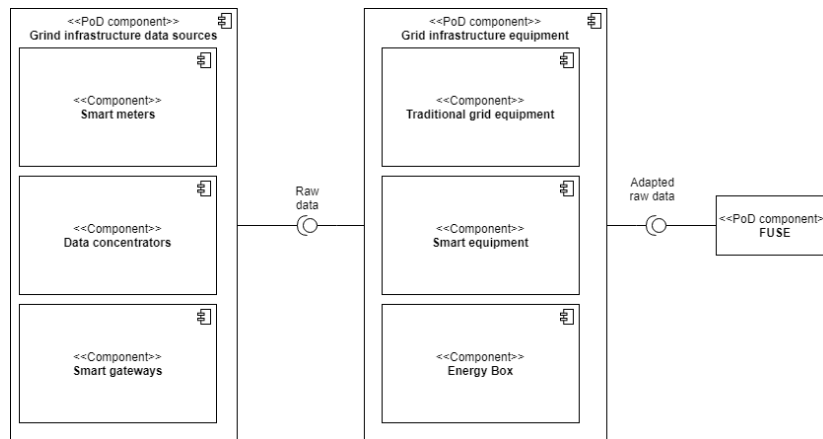


Figure 5. Grid equipment logical diagram

#### FUSE

The FUSE layer's goal is to serve, from one viewpoint, as an interface for outside systems bringing connectors for edge devices and, then again, to give a bound together interface to cutting edge energy services, abstracting them from data management functionalities, like harmonization, mid-to-long term storage or even context broking.

#### Adaptors

The principal motivation behind this module is to build up associations between the energy assets and the FLEXIGRID Information Platform. Contingent upon the technology utilized, a few adaptors are furnished to guarantee smooth communication with FLEXIGRID devices producing data or being controlled in the physical layer. Thusly, all changes made as well as wrappers created in both hardware devices and software modules to communicate with the CIM are viewed as a feature of this module.

So, the usefulness of the device adaptors is to change data originating from both outer hardware and software to guarantee compatibility with the FLEXIGRID CIM. As displayed in Figure 6, a single adaptor gets data produced by energy devices and parses it to incorporate it within the Common Information Model.

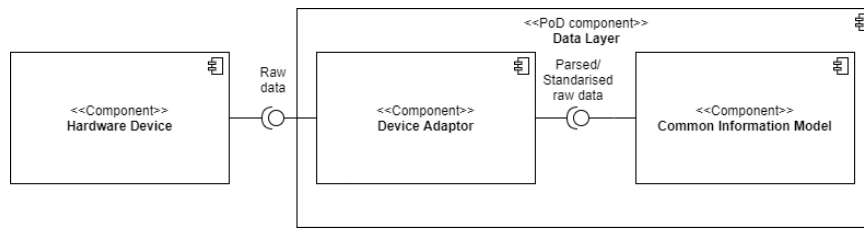


Figure 6. Single adaptor logical diagram

### Common Information Model

The data harmonization of all information got from devices on the field is a responsibility of this module. This harmonization is done dependent on standard and notable ontologies on the ICT and energy domains, bringing about a common structure for data being fed to the principal infrastructure of the FLEXIGRID platform that can be handily charted to these base ontologies. Its motivation is to empower interoperability of the physical layer with the advanced services created for upper layers, or with different services external to FLEXIGRID if required. Figure 7 depicts an illustration of the Common Information model interfacing with device adaptors and other compatible hardware devices and how it harmonizes the data received from them both to arrive at the FLEXIGRID Information Platform.

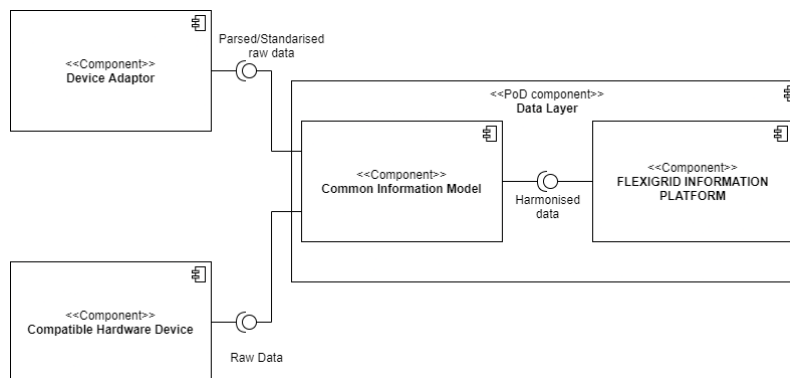


Figure 7. Common Information Model logical diagram

### Service API

This can be considered as an abstraction layer created to orchestrate the manner in which advanced energy services demand data and send commands to underlying energy devices. This API module will guarantee interoperability with FLEXIGRID functional services through coordinating with their required interfaces. The data requests and usage can be, in this way, worked with paying little mind to the underlying intricacy and contrasts in protocols utilized by diverse data sources.

Figure 8 illustrates the interaction among the unified API and the upper layers and underlying components. One way, it gets data requests or device commands from the FLEXIGRID applications by means of the RESTful Web Services and transfers them, while, in the other bearing, it returns the requests coming from the Data Ingestion and Exposition modules (counting visualization or monitoring and alarms, via Security & Access Control module and even the Semantic Repository), which couples with the data present in the Context Broker.

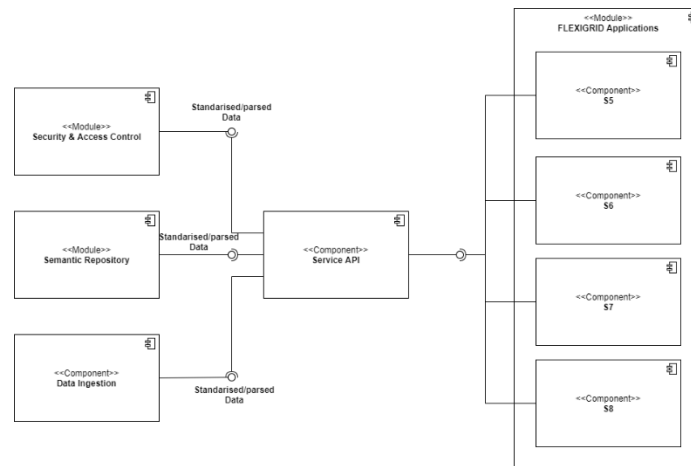


Figure 8. Unified API logical diagram

## Applications

### Fault location and self-healing

The module will determine the section of the network where the fault has occurred, and at the same time prepare a reclosing sequence for the circuit breakers available for this purpose.

### Forecasting and grid operation

Forecast module is a service for providing local consumption and generation predictions over time. This service enables grid operation module to optimize state of the objective function, respecting operational and economical constraints.

### Orchestration service

Since forecast and grid management modules from logical point of view are two serial applications, there has been adopted an Orchestration service, to handle the various needs and communication between above mentioned modules.

### Forecast Module

Forecast module in fact is composed from two distinct modules: one for Photovoltaics production and other one for Load consumption. Each of these two modules are again built by stacking two sub-modules:

- Long term: it provides the forecast for next 24 hours with wide granularity (e.g., 15 minutes). It is triggered and launched on predefined timetable.
- Short term: forecast for only next step, and it is only triggered if in current time step an error above certain threshold has been detected.

Inputs for the sub-modules are as followings:

- PV-long: 3<sup>rd</sup> party weather forecast for days ahead.
- Load-long: 3<sup>rd</sup> party weather forecast for day ahead and device-related database for some days before to present.
- PV-short: data from device-related database.
- Load-short: data from device-related database.

Other input data for the modules are retrieved from the functions and internal calculations.

The results will be fed to Orchestration service to be consumed finally by grid management (optimization) service. The component diagram of the module is shown in Figure 9 below.

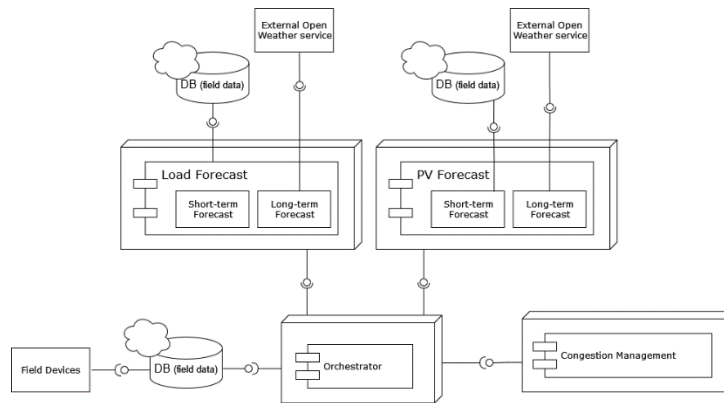


Figure 9. Forecast and grid operation component diagram

### Grid congestion management

This module will utilise the output of the forecast module and real-time and static data from the field to solve overloads and grid issues by optimally managing of energy flow according to the available flexibilities and constraints.

### Grid management (optimization) application

This module is designed and implemented as a hybrid solver, including a metaheuristic and a greedy sub-module which respond to long and short-term forecast accordingly. Therefore, the inputs are as followings:

- Metaheuristics: long time-windows (e.g., 48 hours) scheduling. The inputs are coming from forecast and static data are retrieved statically from local configuration file.
- Greedy: fast decision making for just next step of operation.

The results are fed again to the Orchestration service, to be instructed as the set-points to field controllable devices. The component diagram of the module is shown in Figure 10 below.

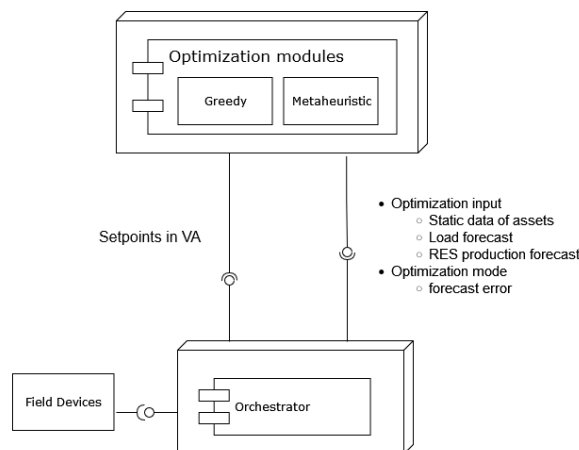


Figure 10. Grid congestion management component diagram

### Thermal energy storage optimization

The thermal energy storage optimization product will be employed in the Croatian pilot site for the quantification of available electricity flexibility from consumers, as well as for the

implementation of Demand Response (DR) requests. The implementation details/view of the constituent modules/components is detailed below.

#### Building Monitoring and Control Dispatch Module

The Building Monitoring and Control Dispatch (BMCD) module is responsible for the data acquisition and storage, as well as for the implementation of the DR requests, i.e., controlling the available heating/cooling assets so as to follow a prescribed consumption curve. It consists of two submodules, the gateway, which is deployed on the consumer premises and performs the low-level communication with the various smart devices, and the Automation Bus, that is hosted on Hypertech's premises and deals with the automation management and data storage and processing.

In more detail, the gateway includes all necessary device adaptors that enable communication with the smart devices, a local cache for temporary storage of data in case of connectivity issues, as well as the Openhab communication client.

The Automation Bus, on the other side, includes the Openhab communication server, the data repository and the automation management component. The latter one is responsible for orchestrating the flexibility estimation process, as well as listening for the DR requests coming from the DSO-side Control Optimizer, translating them to control actions and passing them to the gateway.

The component diagram of the module is shown in Figure 11 below.

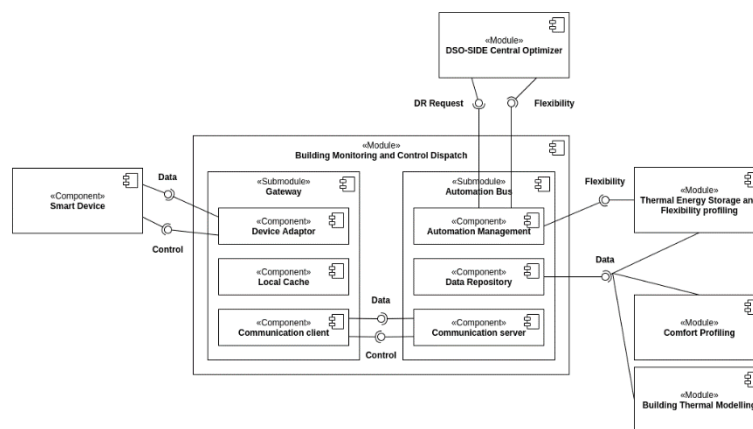


Figure 11. Building Monitoring and Control Dispatch Module Component diagram

#### Thermal Energy Storage and Flexibility Profiling Module

The Thermal Energy Storage and Flexibility Profiling (TESFP) module includes the Flexibility Forecasting component and the Numerical Optimization Engine.

The first component sets up the necessary processes and data for estimating flexibility from an asset. Upon request from the BMCD, the component communicates with components to collect any necessary data and mathematical models, and then triggers the Engine which solves the necessary numerical optimization problems.

The component diagram of the module is shown in Figure 12.

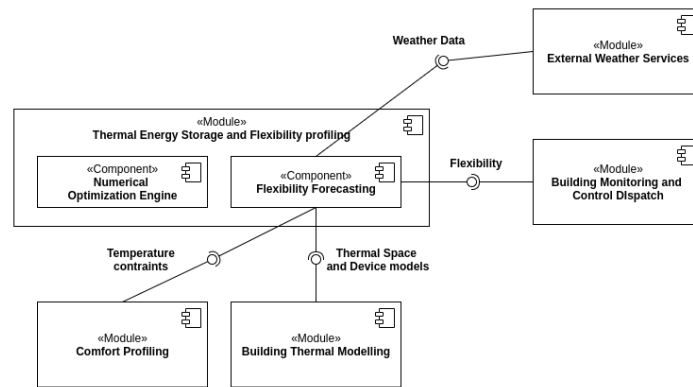


Figure 12. Thermal Energy Storage and Flexibility Profiling Module diagram

### Comfort Profiling Module

The Comfort Profiling (CP) module communicates periodically with BMCD in order to retrieve data from the pilot assets and estimate the comfort profiles of the occupants. The estimation process requires the further estimation of occupancy patterns, extraction of comfort/discomfort events and finally the training of the Naive Bayes Classifier that outputs the profile. Each functionality is performed in a respective subcomponent. The learnt profiles are then stored locally and are sent to the TESFP module upon request.

The component diagram of the module is shown in Figure 13.

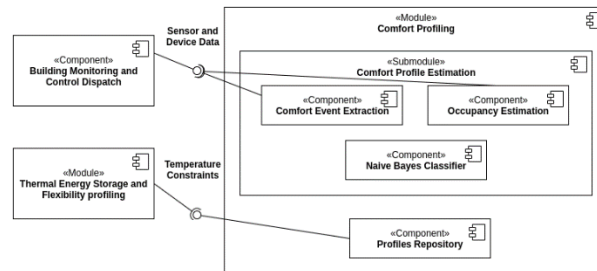


Figure 13. Comfort Profiling Module diagram

### Thermal Model Calculation Module

The Thermal Model Calculation (TMC) module, similarly to CP, retrieves data from the BMCD periodically and performs the identification of the HVAC, space and water heating mathematical models. These models are stored on the repository and are retrieved on request by the TESFV.

The component diagram of the module is shown in Figure 14.

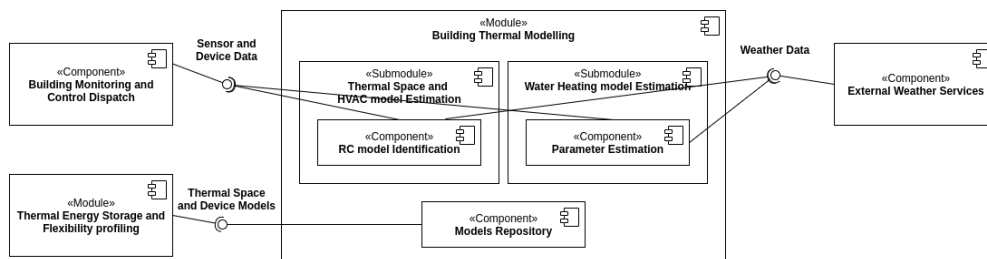


Figure 14. Thermal Model Calculation Module diagram

## 2.5. FLEXIGRID's Reference Architecture. Process View

This view sums up the unique idea of the FLEXIGRID project, as it incorporates the sequence diagrams and particularize them for pilot sites. These sequence diagrams give portrayals of the required collaborations among modules, featuring the message exchanges with their planning and event determinations.

Hence, such sequence diagrams introduced in Annex 3 are the aftereffect of the joint arrangements among pilot partners and technology suppliers. The vast majority of these graphs are self-explanatory, however extra subtleties are given at times to provide a more extensive explanation.

## 2.6. FLEXIGRID'S Reference Architecture. Scenarios View

The view discussed in this chapter, known as "Use Case" or "+1" scenarios view, is utilized to address the use cases according to the perspective of the various stakeholders (integrator, developers, users, and so on). In this specific circumstance, this view targets joining and placing in connection any remaining 4 views, through the functionalities of FLEXIGRID platform that will be converted into concrete requirements.

Thusly, this "+1" scenario view is dedicated to depicting:

- Tools/assets/modules utilized by each pilot site.
- The actors in question and how they cooperate in the FLEXIGRID platform. Functional objectives connecting to the justification for why a given user may act over a specific tool.

The use cases introduced are a mix of the information previously brought in other project reports refined with conversations held at pilot level to particularize and additionally detail the scenarios. This bit of information is further discussed in Annex 4.

## 3. T5.2 Protocols and standards, interoperability and FLEXIGRID CIM

The first part of this task is to study the protocols and standards available in the energy market from a data sharing perspective, and how they are being applied, or are planned to be applied, in FLEXIGRID.

The methodology followed was to first analyse the COSMAG document [12] to identify possible interactions between actors and available standards for those interactions. Then, those interactions were recognized in FLEXIGRID's deployment architecture (Section 4 in D5.1 [1]), and, subsequently, the protocols and data models identified were subsequently reported.

Next step implies the provision of a data model to ensure the interoperable information exchange in the system and that every partner has the necessary variables to perform their defined functionalities. FLEXIGRID CIM data model has been constructed based on the data models included in the CIM standard norms (e.g., IEC 61970-301:2013-12, IEC 61968-11, IEC 62325-301) [2-4]. The main objective of these standards is to facilitate the integration of Energy Management System (EMS) and Distribution Management System (DMS) applications developed independently by different vendors.

### 3.1. Protocols

The protocols have been divided in two sections:

1. Protocols used for data collection in the field. These protocols apply to communication between appliances and the control centre. These appliances can be smart meters, Intelligent Electronic Devices, electrical substation, power plants, etc... Some of these protocols are also used to collect meteorological data from weather stations.
2. Protocols used for data sharing. These protocols are used for exchanges between platforms, mainly between FUSE and other platforms belonging to the pilots (VIESGO, Hypertech's cloud, ALPERIA's platform). In the case of the HTTP, the protocol is also used to share data from wheatear providers, CIRCE's Energy Box and smart meters from the Italian demonstrator.

#### *Protocols used in FLEXIGRID for data collection in the field*

##### *IEC 101/IEC 104*

The set of standards IEC 60870 defines the use of telecontrol equipment and power systems and is developed by the IEC Technical Committee 57. Within those standards, the communication protocol is defined in part 5 (IEC 60870-5) which, in order to implement its specifications, one must apply one of the profiles defined in *companion standards* IEC 60870-5-101 to IEC 60870-5-104, also called IEC 101 to IEC 104 for short. [13]

The standards briefly described here, IEC 101 and IEC 104, are widely used in the energy sector and can be thought as being fundamentally the same. While the former defines basic user functionalities for telecontrol tasks, the latter uses a combination of the application layer of IEC 101 and functionalities of the TCP/IP transport layer [14]. This is illustrated in their protocol stacks, shown in Figure 15.

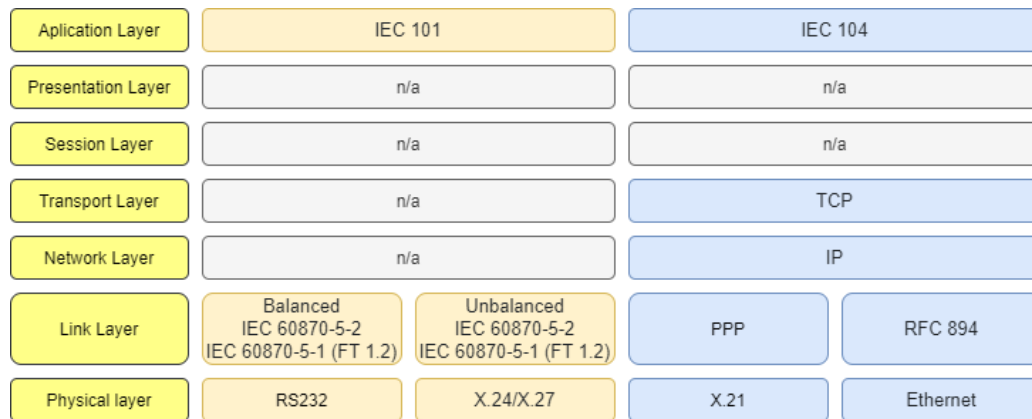


Figure 15. IEC 101 and IEC 104 protocol stacks (OSI Model). Image based on the information available in [15]

The application layer, defined in IEC 101, specifies the data structure of messages to be transmitted for control and monitoring operations between a control centre and an RTU (Remote Terminal Unit). These messages, called ASDUs (Application Service Data Unit) can be of different types, *including all a common header. Depending on the type of ASDU, the messages sent can contain process information, system information, control direction parameters or files.* [16, 17]

*Even though these protocols are not explicitly mentioned in the COSMAG document, they are described in deliverable D3.2 of the project IDE4L [18], which is acknowledged by the COSMAG as a reference document for internal DSO data flows, especially for power systems automation.*

In FLEXIGRID, only IEC 104 is used both in the Spanish and the Croatian demo site to communicate IEDs (Intelligent Electronic Devices) with the DSO's SCADA systems.

#### IEC 61850

IEC 61850 is the latest standard for communication networks and systems in electrical substations, defining a wider scope than IEC 101, its predecessor.

Unlike Modbus and IEC 60870-5 protocols, IEC 61850 was intended to work over IP networks since its conception (although this is a consequence of being a more contemporary standard). This can be acknowledged in its protocol stack, shown in Figure 16. Also, note that IEC 61850 supports mappings with a variety of application layer protocols (i.e.: SV, GOOSE, GSSE and MMS) and uses SNTP for clock synchronisation.

Application Layer	TimeSync SNTP	SV Sampled Values	GOOSE Generic Object Oriented Substation Event	GSSE Generic Substation Status Event MMS ISO 9506 Connectionless ACSE ISO/IEC 8649,10035	MMS ISO 9506 Core ACSE Services Connection-oriented ACSE ISO/IEC 8649,8650	
Presentation Layer	n/a	n/a	ASN.1, BER ISO/IEC 8824.1	Connectionless presentation ISO/IEC 8649,10035 ASN.1, BER ISO/IEC 8824.1	Connection-oriented presentation protocol ISO/IEC 8822,8823 ASN.1, BER ISO/IEC 8824.1	
Session Layer	n/a	n/a	n/a	Connectionless session ISO/IEC 9548	Connection-oriented session ISO/IEC 8326,8327	
Transport Layer	UDP/IP	n/a	n/a	GSSE T-Profile ISO/IEC 8602	ISO CO T- Profile ISO/IEC 8073	TCP/IP T- Profile ISO Transport on top of TCP (RFC 1006)
Network Layer	IP (RFC 791)	n/a	n/a	ISO/IEC 9542	ISO/IEC 8473	IP (RFC 791)
Link Layer	RFC 894	Priority Tagging/VLAN (IEEE 802.1Q) CSMA/CD (ISO/IEC 8802.3)		ISO/IEC 8802-2 LLC		RFC 894
Physical Layer	ISO/IEC 8802.3 Ethernet			ISO/IEC 8802.3		ISO/IEC 8802.3 Ethernet

Figure 16. IEC 61850 protocol stack [15]

Moreover, IEC 61850 offers the possibility of using an oriented event communication scheme apart from master/slave, which it also defines. These event-oriented communications allow fast, reliable multicast message exchange (point to multipoint). The message transferred are called GOOSE (Generic Object-Oriented Substation Event). [17]

Another technical improvement that has been observed over previously released standards is that the data transmission rate is faster, as shown in Table 1, meaning that the execution of a command can be accomplished in less time. [19]

Table 1: Comparison between some protocols used in system automation in electrical substations [20]

Protocol	Data Rate (M bits/s)
IEC 60870	0.19
Profibus FMS	12
DNP V3.00	0.12
Modbus	0.12
UCA2	100
IEC 61850	100

Additionally, apart from defining the transmission format, IEC 61850 also defines a hierarchical abstract information model that reduces time searching for information, and a language for

describing devices and configurations in a vendor-independent manner, promoting interoperability. [17]

The COSMAG points to IEC61850 as an automation protocol and data model for substations, which only concerns the DSO. However, it is also mentioned as a solution available for the interface between the DSO and a prosumer, which is described as “still under development”, meaning that it has not been fully established yet.

In FLEXIGRID, this protocol is used for data communication in the Italian demo site to transfer data between MV and HV buildings (i.e., electrical substations, power plants and MV customers) and the SCADA system, deployed in ALPERIA’s platform. It is also used to communicate remote terminal units located in MV power plants in islanded operation mode.

#### DLMS/COSEM

DLMS/COSEM is a client/server protocol used in smart energy metering, management and control that has been published by the IEC as an international standard under the code name *IEC 62056*. It specifies both an interface model and communication protocols for exchanging data with metering devices. As illustrated in Figure 17, it can be used for several applications, utilities, market segments and communication media. [21]

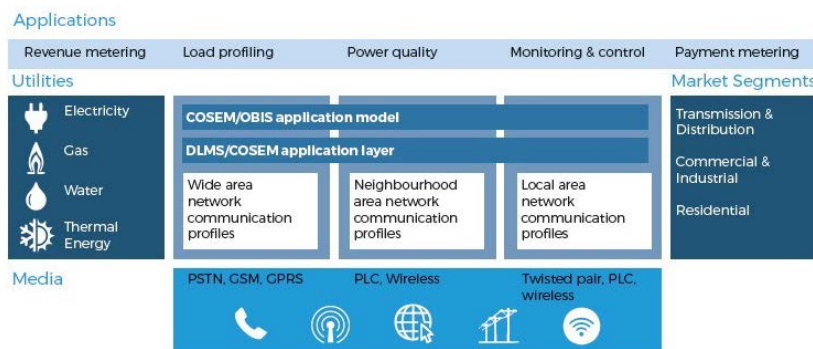


Figure 17. DLMS possible uses and implementations [21]

DLMS/COSEM’s data exchange main properties include: [22]

- Multiple-party access to metering devices and access control to their resources
- Guaranties of protection and privacy by means of encryption
- Selective access, compact encoding and compression, which results in low overhead and efficiency
- Possibility for configuring a single access point when multiple metering devices are available in the same site
- Possibility to implement both local and remote data exchange simultaneously
- It can be implemented over various communication channels, within many spatial scopes (LAN, NAN, WAN)

From an implementation perspective, the protocol is composed of three main components. First, COSEM (Companion Standard Specification for Energy Metering), which is an object model that describes the semantics of the language. Then, OBIS (Object Identification System), which is the naming system for COSEM objects, which is specified for metering utilities (electricity, gas, water, etc.), but is also defined for abstract measurements not related to the energy market. And last, DLMS (Device Language Message Specification), which defines the syntax of the

language. Accordingly, COSEM would be located in the OSI presentation layer, while DLMS would be in the application layer, as illustrated in Figure 18. [21]

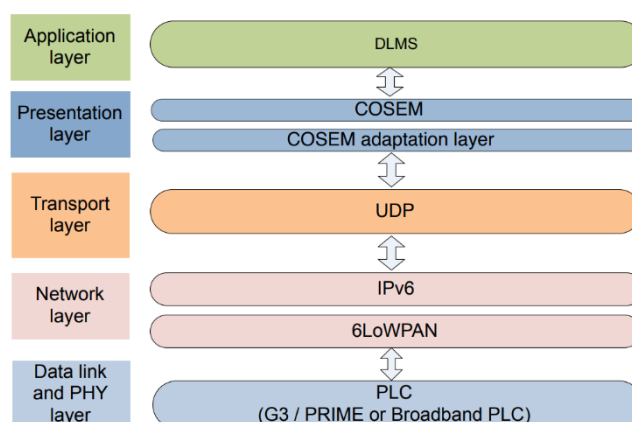


Figure 18. DLMS/COSEM protocol stack [23]

The COSMAG identifies DLMS/COSEM as one of the possibilities for the interaction between prosumers and retailers. However, as a consequence of the aforementioned characteristics, it can also be used for data collection by other actors (e.g., DSOs, ESCOs or customers themselves). Indeed, DLMS/COSEM is being used in FLEXIGRID for collecting data from smart meters in LV customers in the Spanish demo. Additionally, DLMS is used in the Italian demo as a back-up communication protocol for smart meters, using radiofrequency.

### Modbus

Modbus is a level-7 OSI messaging protocol that provides master/slave communication among multiple devices connected through buses or networks. It is mainly used for data collection and for the possibility that it offers for managing different devices.

Since its original publication in 1979 as an open standard, Modbus has become a de facto standard within the industry for its relative simplicity in data representation and ease to be deployed [24]. Consequently, nowadays many devices are compatible with it in the energy market.

At present, as illustrated in Figure 19, Modbus is implemented in several ways, including TCP/IP over Ethernet; asynchronous serial transmission over diverse channels, such as RS-422, fibre, or radio; and MODBUS PLUS, which is a high-speed network version of the protocol that uses token passing. [25]

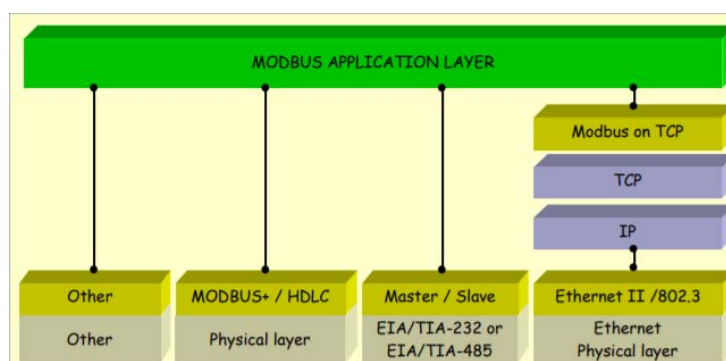


Figure 19. Modbus communication stack [25]

In FLEXIGRID, Modbus is used in many demo sites:

- In the Spanish demo, it is used to send meteorological data from weather stations and for communicating with the smart equipment of the new substation of the future. It is implemented using the TCP/IP stack in both cases.
- In the Greek demo site, it is used to locally communicate all devices within each building (bungalows and substation). It is implemented using the asynchronous serial transmission over RS-485 stack
- In the Italian demonstrator, it is used for internal communication within each substation, power plants and MV customers. It is mainly implemented using the TCP/IP stack, but the implementation can vary depending on the building.

#### *MQTT*

MQTT (Message Queuing Telemetry Transport) is a standard messaging transport protocol based on a publish/subscribe paradigm. It was first published in 1999, but it has recently been flourishing again for M2M communication and IoT applications due to its simplicity, lightness, scalability, and the fact of it being an open OASIS and ISO standard (ISO/IEC 20922). Its usual protocol stack is the same as any other application layer protocol running over TCP/IP networks, but it can also run over other ordered, lossless, bi-directional networks (e.g.: Zigbee). [26]

The publish/subscribe paradigm requires that an MQTT broker acts as a dispatcher between MQTT clients, which can act as both publisher and/or subscribers. The standard defines three qualities of service for delivery: At most once (best effort), At least once (duplicates allowed), and exactly once (message arrival assured exactly once). [26]

As a protocol suitable for IoT applications, it can be used for collecting data from small sensors deployed in a defined area and connected to the same network (e.g., sensors deployed in the premises of a prosumer). In FLEXIGRID, MQTT is used by VERD's SmartBox for publishing the data generated in all buildings of the Greek demonstrator to VERD's platform.

#### *Z-Wave*

Z-wave is a wireless protocol for meshed networks, with a primary focus on residential automation (monitoring and controlling home lighting, temperature and security). It is a proprietary standard. However, in 2016 it made its interoperability layer publicly available to allow communication between devices from different manufacturers. [27]

Z-wave protocol stack can be seen in Figure 20. Its physical and link layers are based on the ITU G.9959 specification. The transfer layer provides communication between neighbouring nodes, acknowledgement of packets, node awakening and packet origin authentication. The routing layer, as its name implies, handles the routing of packets, including routing table updates and topology scan. Lastly, Z-wave application layer manages the payload received or transmitted. [28, 29]

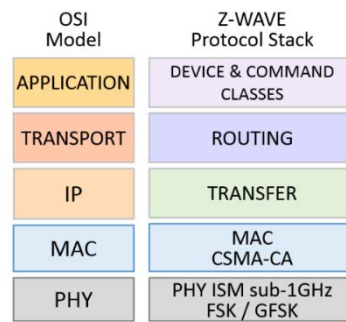


Figure 20. Z-wave protocol stack [28]

As it has been widely accepted as a home automation protocol, Z-Wave can be used for collecting data in end users' premises. As a matter of fact, an upgraded version of it called Z-Wave Plus will be used in FLEXIGRID to collect data from LV and MV consumers in the Croatian demonstrator. Such upgrade includes an extended communication range, more bandwidth, self-healing and plug-n-play functionalities, among other improvements [30].

### *Protocols used in FLEXIGRID for data sharing*

#### *AMQP*

AMQP (Advanced Message Queuing Protocol) is an open OASIS protocol for transferring business messages. Similar to MQTT, it is an application layer protocol that runs over TCP/IP networks, making its protocol stack no different than that of a typical internet suite application.

Internally, AMQP is defined in several layer. The lowest layer is a binary, peer-to-peer protocol for transferring messages between two processes on a network. On top of that, a messaging layer specifies an abstract message format, with particular standard encodings. [31]

Even though AMQP and MQTT are both application layer messaging protocols, their intended use is different. MQTT is aimed for devices sending small messages on networks with low bandwidth, while AMQP is intended to provide richer scenarios, permitting classic messaging queues, round robin, store and forward and any combinations from it. [32] In FLEXIGRID, AMQP is planned to be used to exchange data between Hypertech's cloud and FUSE platform in the Croatian demo site.

#### *FTP*

FTP (File Transfer Protocol) is an application layer protocol designed for transferring files over a TCP/IP network. The current specification for FTP was published in 1985 in RFC 959, which obsoleted the version published in RFC 765 [33].

FTP uses a client-server paradigm and uses different channels (ports) for control and data transfer and it has two modes of operation that determine how the connection for the data channel is established: active or passive. In active mode, the client opens a port for the server and informs it which one is it. Afterwards, the server uses that port to initiate the data connection. In passive mode, the server has reserved a range of ports for data communication, which it indicates to the client upon request so that the client can initiate the data connection. The latter mode has a more extended use nowadays due to clients being usually located behind a firewall throughout the Internet.

In FLEXIGRID, FTP has been selected to be used for sharing data between VIESGO's platform and FUSE, being this solution the most appropriate given VIESGO's security policy. However, FTP is

not considered a secure protocol (it lacks encryption). This means that at the moment of implementing this communication, the parties involved will have to choose between currently available secure alternatives for FTP, such as FTPS (FTP over TLS) or SFTP (an alternative protocol that extends SSH).

#### HTTP

HTTP (Hypertext Transfer Protocol) is an application layer protocol to transfer hypermedia information. Its latest standardised version is HTTP/2, published in 2015 in [34], but there is an upcoming version of the protocol that uses QUIC as a transport layer protocol instead of TCP and whose standard is currently being drafted [35].

The reason why HTTP is mentioned in this document is because FUSE provides RESTful web services using HTTP to let other devices or platforms connected to the Internet to update or retrieve the context information to the cloud. Currently, FUSE implements these interfaces using NSGiv2 in a FIWARE-powered cloud solution.

Specifically, the data connections that are planned to be made using RESTful web services in FLEXIGRID are:

- For providing context information related to the secondary substation of the future, and for CIRCE's Energy Box to update and retrieve data to FUSE in the Spanish demonstrator.
- For VERD's SmartBox and CIRCE's Energy Box to update and retrieve context information, for getting information from external weather providers using adaptors, and for possibly receiving commands to control field devices.
- For exchanging context information between HEP-ODS's platform and FUSE, and for getting information from external weather providers using adaptors in the Croatian demo site.
- For getting the concentrated data generated by smart meters from a central data warehouse in the Italian demonstrator.

#### OPC

OPC (Open Platform Communications) is an open communication standard for secure and reliable data exchange in the industrial automation sector and IoT. It was first published in 1996 as a proposition to abstract PLC specific protocols (e.g., Modbus) into a standardised interface. This way HMI and SCADA systems would interact with an intermediary using generic OPC request that converted them into device-specific requests and vice versa. [36]

Originally, the standard was limited to Windows Operating system and thus functioned over a COM/DCOM presentation layer (OPC Classic). However, with the establishment of service-oriented architectures in manufacturing systems, the OPC Foundation decided to create a successor (OPC UA, which stands for OPC Unified Architecture) that was platform independent and has interoperability as its main objective. As a result of this, Linux and on-chip implementations exist nowadays, being able to use OPC UA from sensors to the cloud. The protocol stack for OPC UA is shown in Figure 21. [15]

Application Layer	UA Application	
Presentation Layer	UA Binary	UA XML
Session Layer	UA TCP, UA Secure Conversation	SOAP/HTTP, WS-SecureConversation
Transport Layer	TCP (RFC 793)	
Network Layer	IP (RFC 791)	
Link Layer	MAC (IEEE 802.3)	
Physical Layer	Ethernet (IEEE 802.3)	

Figure 21. OPC UA protocol stack [15]

In FLEXIGRID, OPC UA is used to exchange data between ALPERIA's platform and FUSE in the Italian demo site.

### 3.2. Data Models

In order to allow different vendors to participate in a common project where data will be shared, a common data model is essential. Indeed, there would be information coming from different appliances and platforms such as smart meter, power plant, weather platforms, etc. Moreover, the information sources are in four different countries (Greece, Italy, Slovenia, Spain) which creates language differences. In this section, we analyse the different data model standards that could help us to create a common ontology.

#### Common Information Model (CIM)

The Common Information Model is a standard that allows the software representation of the elements that constitute the infrastructure, management and operation of electrical power systems.

In the 90s, there was a vendor lock-in originated in the utility marketplace as a result of Energy Management System (EMS) vendors developing their applications using proprietary data models and interfaces. Consequently, it was necessary to make substantial investments in time and money to purchase and support EMSs. Therefore, the CIM was conceived as proposal to solve this vendor lock-in and was adopted as an international standard by the IEC in 1996. It is formally described using UML, and its elements could be of a physical nature (e.g.: devices) or could be abstract (e.g.: A customer agreement). [37]

The CIM is standardised in various series of standards: IEC 61970, IEC 61968, and IEC 62325. Following is a short list of the most relevant parts of these standards that are applicable to this analysis:

- **IEC 61970-301:** It was the first release of the CIM, defining the base for a common semantics for Energy Management Systems. It includes several core components (wires, transformers, switches, etc.). [38]
- **IEC 61968-11:** It defines an extension of the CIM for electrical distribution networks management and for integrating enterprise-wide information systems. [39]
- **IEC 62325-301 and IEC 62325-351:** They specify the CIM for energy markets. Part 351 contains a new set of classes and relationships that comply with European-style markets and regulations [40, 41]

The COSMAG document recognises the CIM as a key standardisation element for exchanging data within the scope of the DSO's operations, but it also mentions that efforts have been made

towards adapting other actor interactions, such as the one between aggregators and prosumers, to map an available standard for demand response (i.e., OpenADR) to the CIM.

Additionally, the COSMAG document also proposes an extension of a SAFER-based data modelling base to support the contexts present in the CIM.

### FIWARE

As one of the European Commission's flagship projects, the FIWARE program develops an open-source framework for building IoT applications. It aims to facilitate the development of smart applications across diverse sectors by providing unified, royalty-free, implementation-driven software platform standards. [42]

It consists of a set of modular tools, called *Generic Enablers* that provide different functionalities and that can be integrated with each other by means of leveraging from a unified interface.

The core and mandatory component of such generic enablers is the *FIWARE Context Broker*. It provides the resources for managing context information and for integrating the other optional generic enablers. Among these resources is the aforementioned interface, which currently matches NGSIv2 specifications [43]. Nonetheless, it is evolving to match NGSI-LD, which is currently a standard adopted by ETSI Industry Specification Group [44]. NGSI-LD defines an abstract information model to manage context information that spreads across multiple application domain.

The optional generic enablers provide functionalities for: [42]

- Interfacing with IoT systems (IoT Agents), robotics and third-party systems
- Processing, analysing and visualising context
- Managing context data and APIs, Including cybersecurity capabilities (e.g., access control), publication and monetization

Additionally, as part of the framework and interoperability environment developed by FIWARE, it includes the definition of basic data models to enable data portability across different applications, such as Smart Cities, Smart Energy and Smart Buildings, among others. [45]

The COSMAG document indicates FIWARE as a feasible solution to serve as a bridge between smart grid applications and smart city platforms or other domains. It specifies that this should be done by providing an adaptation to cover the context presented by the CIM and proposes to do so by extending the SAREF. It also mentions that FIWARE could potentially integrate proprietary solutions, thereby removing data silos. [12]

In accordance with that, it is worth mentioning that FUSE is a powered-by-FIWARE platform that uses SAREF as one of the base ontologies for the definition of its data model.

### OpenADR

Historically, electric companies produced power according to customer demand, increasing or reducing production at each moment in power plants and buying or selling energy to other companies. However, these solutions are limited. The concept of Demand Response revolves around the idea of adapting the end-user's power consumption, increasing or reducing it in response to, for example, dynamic pricing or peak demands. This way, it is possible to achieve more efficient use of the power resources [46].

Furthermore, Distributed Energy Resources (DERs) play an important role in this, as they can be used as an alternative power source for the customer. Electric vehicles or supplement grid-scale storage are also observed as they introduce new power demands in the electric grid.

In accordance with this, there was a need for an open standard that allows the exchange of DR messages, introducing OpenADR (Open Automated Demand Response).

OpenADR is a smart grid standard that defines an open, secure, bidirectional information exchange model for demand response applications. It describes the format of messages used for automated demand response and DER management, allowing the exchange of dynamic price and reliability signals in a consistent and interoperable manner. Moreover, its OpenADR 2.0b Profile Specification has been approved by the OEC as a Publicly Available Standard (IEC 62746-10-1). [47, 48]

The COSMAG identifies OpenADR as an important element to consider in the communication between aggregators and prosumers, mentioning that there have been efforts to map it to the CIM and to integrate its approach with SAREF.

#### *SAREF*

SAREF (Smart Appliances REference ontology) is a standard ontology adopted by the ETSI that defines a common terminology and the relationships existing between entities in the smart appliances' domain.

SAREF was created based on the following principles: [49]

- **Reuse and alignment:** Concepts and relationships already defined in other standards have been harmonized and aligned (e.g.: W3C® SN ontology, UPnP®, OM Ontology of units of measure)
- **Modularity:** It allows to separate and recombine different parts of the ontology to match specific needs
- **Extensibility:** Enabling growth of the ontology, specialising SAREF concepts concerning the needs of different stake holders or adding support to external domains. Currently available SAREF extensions include SAREF4ENER (extension for the energy domain), SAREF4ENVI (extension for the environment domain) and SAREF4BLDG (extension for the building domain)
- **Maintainability:** To ease the process of detecting and rectifying flaws.

As already specified, the COSMAG document considers SAREF as an important candidate for a base ontology that can be extended to contain the complete energy value chain.

### 3.3. Protocols and standards

#### *Representation Standards and FLEXIGRID Project*

As task leaders of the architecture and data modelling tasks, ATOS and CIRCE begun the preliminary communications to establish the work methodology along this job. Supported by the study performed in the early stages of this task and reflected in D5.3 *“Protocols and standards definition”*, the initial steps to select the most appropriate standard for the architecture were directed to the analysis of the most advanced tool so far in the project, that is the FUSE platform from ATOS.

FUSE is a powered-by-FIWARE platform that uses SAREF as one of the base ontologies for the definition of its data model and aims to be an open-source platform that enables edge device integration by fully exploiting available data from local and distributed energy resources to build value-added services for DSOs and energy stakeholders.

Familiarization with FIWARE technology was essential at the beginning to integrate the rest of the developments with the platform. FIWARE terminology had to be integrated with the modelling of the information proposed by other standards, particularly those from the CIM, IEC 61968 and IEC 61970. A compromise between the rigidity of the CIM normative and the extreme flexibility of FIWARE was mandatory, that will lead to a merger of the selected data model and FIWARE-NGSIV2 specification to keep compatibility with the data ingestion module, as seen in Figure 22.

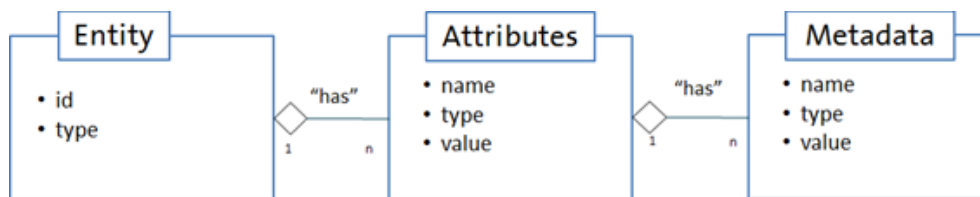


Figure 22. Main elements in the NGSI data model

FUSE provides a context broker which harmonizes data from different providers and oversees the parsing process of the resulting JSON format of the data model. Being able to understand the information format exchanged in JSON files is an essential feature from FUSE, that allows the data flow between its architecture layers. For this reason, the model presented here uses JSON as information interchange format.

ATOS provided the partners with a preliminary data model as an example of the FIWARE restrictions, that was already mapped with the SAREF ontology. Using this model as a reference, CIRCE started working on defining the data representation standard to model the project information that best fits the needs of the project and adapts the data model to the platform powered by ATOS. SAREF and FIWARE data models were used as a basis to be extended to the rest of the components using the FLEXIGRID CIM context.

From the study performed in D5.3 [3] of different solutions for gathering information and exchanging it through the system, CIM standard was chosen as a data model solution for several reasons. First, it allows the representation of common elements of electrical power systems and it is one of the core standards of the future smart grid as pointed out by different organizations and recent standardization roadmaps. Its key purpose is to provide a common language to describe exactly what data is being exchanged among different business systems. Its features can help with standardized semantics regarding the representation of electrical grid topologies. All these characteristics provide ease and interoperability along the architecture.

Considering that this option must be interoperable with the FLEXIGRID applications, a common format to transport standard CIM entities is required. The selected format for messaging among FLEXIGRID applications was JSON to easily deploy applications in a FIWARE context. This approach allows to take advantage of the grid modelling tools and integrate them smoothly in standard CIM agnostic systems. This mechanism is described in the next sections.

### *FLEXIGRID Common Information Model design guidelines*

The data model provides a common vocabulary to be used by all project partners, a common way for modelling the distribution grid under the CIM standard (in the form of .XML files) and transmitting and exchanging the associated data (in JSON format). To ease the work for all the partners and facilitate the understanding of the data model to all of them, some design criteria was established in that sense.

The standard to be used is the static CIM (IEC 61968) as a grid topology description. This will allow to take advantage of the existing entities in the normative, instead of creating a whole new abstract model. The entity names defined in the CIM standard will be reused, and if needed entities outside the normative could be defined.

Configuration data related to the field devices and grid topology is out of the scope of this task, the complexity and abstraction added to the model with this information makes the model infeasible to use.

It is important to stress the scope of the data model described in this document. This task is limited to the modelling of the information needed by the software solutions and exchanged among the different field devices and software modules in the FLEXIGRID architecture. Not all the information contained in the standard CIM files generated by the grid design tools will be detailed in this document.

For this reason, in the JSON templates there will be a reference to the CIM identifiers of each needed entity and only measurements and source will appear. This way the integration between XML files of the CIM standards and the JSON format used to communicate with FUSE is achieved.

There will be two different identifiers, one that keeps the semantic meaning of the instance, and other human-readable. Also, no measure is going to be mandatory in JSON instances to avoid problems with missing quantities, guaranteeing that only the minimum information is mandatory for the integration with FUSE.

The development of data models usually entails to deal with a great level of abstraction, translating information provided by physical devices and variables into a complex format defined by a standard. This process usually involves relevant difficulties and challenges to be addressed and solved.

In the FLEXIGRID case, a holistic approach was carried out, trying to cover all the possibilities that could arise in the different scenarios. This holistic approach increases a lot the complexity of the earliest steps of the process but also the adaptability to changes during the project. Once the most elaborate and ambitious version was presented and having a better knowledge about the protocols and the needs of each demonstrator site, a simplified version was developed to cover the current needs expressed by each of them.

The initial model based on FIWARE provided only entities referring to field devices with a load profile, but other roles were needed to consider. Entities for generation, transformation and storage were introduced to represent the basic FLEXIGRID entities as can be seen in the next section. The use of the CIM standard will solve the definition of many required entities that will be included in the FLEXIGRID CIM.

As mentioned before, the great complexity of the CIM standard could be an issue to achieve simple interoperability among the different software modules of the project. That complexity will be addressed by employing only the key entities used by the FLEXIGRID software solutions.

### 3.4. FLEXIGRID Common Information Model

#### *Methodology to define entities required for the FLEXIGRID CIM*

All demonstrator sites have been considered in the production of the FLEXIGRID CIM, but the Spanish demonstrator was selected as the model to start developing it. Integrating the rest of the demonstrators was easier once taking it into account the most complex of them. The protocols and standards studied in D5.3 “*Protocols and standards definition*” were also taken into account.

Several meetings were held with ATOS, as leader of WP5, and it was decided to finally use the CIM standard for its semantic interoperability. A first version of the FLEXIGRID CIM was sent to ATOS with the information from the demos with a proposal to map the CIM standard to JSON files and use them as an information exchange format with fully integration into FUSE platform. Once validated, it was agreed to send it to the partners. Any further modification in the FLEXIGRID CIM to accommodate the demos will be performed in T5.4.

One of the first steps of this task was the definition of a methodology for the partners to develop the common data model in a collaborative way, to establish a roadmap with the appropriate milestones. The definition of a common model of data exchange is always a challenge, especially when there are several partners involved.

In order to have a complete perspective of this situation and to better understand how the FLEXIGRID CIM was designed, a Venn diagram has been prepared, as shown below in Figure 23.

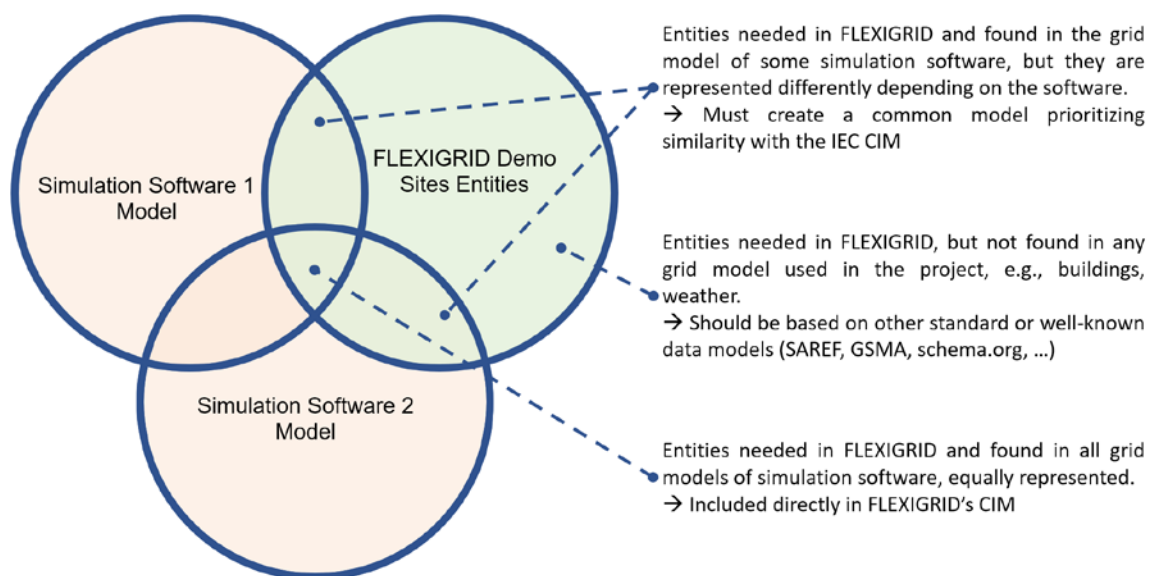


Figure 23. Methodology to define the entities of the FLEXIGRID CIM

According to Figure 23, the FLEXIGRID CIM only refers to a sequence of entities that create all the intersections and are those that are required for any software that exports to CIM. The proposal here is to define only the needed list of entities of the FLEXIGRID CIM and not to query

the entire structure of the CIM standard classes. Therefore, the model must be unique for each demo site. Assuming that the formats used by other demonstrator sites are not exactly the same as DigSilent (for example, attributes that describe the same characteristic could be named differently), within FLEXIGRID a compromise would have to be reached to carry out a conversion between the two.

Each demonstrator will export the CIM from its own software tool, forcing to establish some minimum requirements in the model that each of them will produce to make it understandable and interoperable. These requirements are a common topology and a list of referenced entities. That is why the intersections of the Venn diagram indicated in Figure 41 allow us to know which entities in the FLEXIGRID CIM demonstrators will needed to be in common. Those demonstrator sites which do not have grid modelling software will use the information (.JSON files) corresponding to the intersection of the Venn diagram, which turns out to be the minimum information necessary.

The first step was to define the list of entities of the intersection which define the intersection shown in Figure 23. In the case of the Greek demonstrator, no grid simulation software is used and therefore no files are generated under the CIM standard. So, these entities, according to the above diagram, will correspond to the intersection, which is the minimum needed. In consequence, as seen so far, there are entities that are not related to the grid and other entities that are generated with grid simulation software. Some entities have been used in the final version of the FLEXIGRID CIM and others that are not. There was a need to reach a compromise to model the entities in the FLEXIGRID CIM so that the demonstrators can use them.

The key to this methodology is to guarantee semantic interoperability. Those demonstrator sites that do not have their modelling software, will work with these entities in the form of .JSON files, so that they can model their grid and can also import it into any modelling software. That is why we need the entities defined in the FLEXIGRID CIM to have the minimum attributes necessary to create them, and that due to semantic interoperability all this information can be loaded from a grid topology modelling software.

The preliminary version of the FLEXIGRID CIM, based on common grid elements, is explained in Figure 24. It shows several commonly identified entities useful for the information exchange needed for FLEXIGRID functionality. They are arranged in groups with same capabilities called *Categories*. Categories are meant only as a classification container, but they do not have explicit representation within the frame of the FLEXIGRID CIM. The rest of the entities should have description within the model described here and a JSON representation will be provided for interoperable communication purposes.

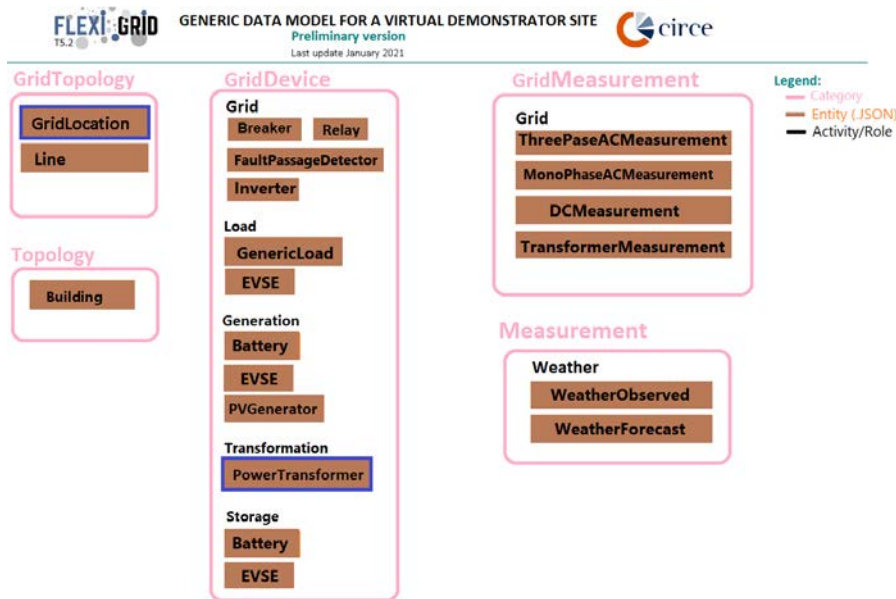


Figure 24. Common grid entities

Each category represents a different level of the grid model. On the top we have *GridTopology*, gathering the different possible positions of an element in the grid. Next the *GridDevice* represents different elements currently inside each location. Finally, those devices usually can gather information, which is represented by the *GridMeasurement* entities showed above.

The *GridTopology* reference will be used by the FLEXIGRID software solutions to discover the origin of the measurement and the underlying grid topology. Therefore, they will be able to locate each device. Most common grid modelling software can export the grid topology using the CIM standard format in the form of separate .XML files. Below it can be found a definition of the different entities contained in the FLEXIGRID CIM:

- **GRIDLOCATION:** represents a location within the grid (the electrical grid is represented by node voltages); boundary points are included within the grid location.
- **LINE:** a set of conductors used to transmit and distribute electrical energy
- **BREAKER:** a device that save and switch open/closed status.
- **RELAY:** is an electrically operated switch device that has voltages and currents and is used to protect electrical circuits from overload or faults. A relay plus a breaker together form a protection.
- **FAULT PASSAGE DETECTOR:** makes it easier to locate faults on distribution grids.
- **EVSE:** electric vehicle charging station.
- **GENERICLOAD:** load that have active and reactive power.
- **PVGENERATOR:** wherever there is a load, it has both load and generation profile.
- **INVERTER:** an inverter neither generates nor consumes; transforms alternating current (AC) into direct current (DC) and vice versa, so it can be connected to any other equipment to connect it to the grid.
- **BATTERY:** in the CIM standard, this device appears as generator and load at the same time.
- **POWERTRANSFORMER:** is a static machine used for transforming power from one circuit to another without changing frequency. In the FLEXIGRID CIM is considered as an ideal transformer so that it does not modify the load flows.

**THREEPHASEACMEASUREMENT:** represents a measurement from an electrical system that uses three-phase alternating current.

- **MONOPHASEACMEASUREMENT:** represents a measurement from an electrical system that has a single phase and alternating current.
- **DCMEASUREMENT:** takes place in those devices where the flow of electrical current occurs in only one direction.
- **TRANSFORMERMEASUREMENT:** measure resulting from transform a system of alternating voltage and current into another system of voltage and current usually of different values and at the same frequency.
- **WEATHEROBSERVED:** represents an observation of weather conditions at a certain place and time.
- **WEATHERFORECAST:** this entity contains a harmonised description of a weather forecast.

The intersection between these three circles representing FLEXIGRID CIM and all the modelling software used as an example in the Venn diagram is shown in Figure 25 and would be formed by the entities shown in Figure 24.

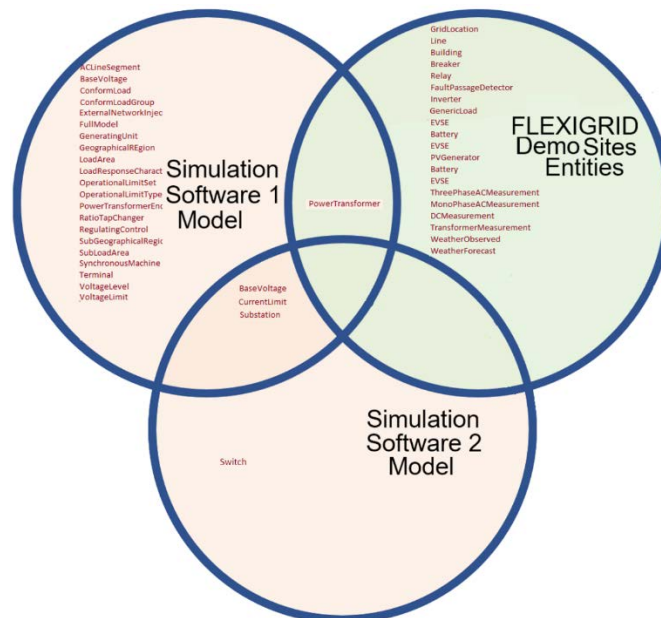


Figure 25. Preliminary list of FLEXIGRID CIM entities

This methodology starts with the modelling of the Spanish demonstrator grid architecture. With the model, the measurements needed to be captured and the nodes of the grid that produce them were identified, as shown in Figure 26.

After working with grid modelling software solutions in demonstrators, these steps are identified to get the entities under the CIM standard:

- Modelling of the demonstrator grid
- Identifying which measurements were needed to be captured and at which points in the grid they will be produced, as seen in Figure 23
- Export this topology under the CIM standard format in the form of .XML files
- From the output .XML files, extraction of the grid topology, obtaining the possible additional entities needed by the FLEXIGRID CIM

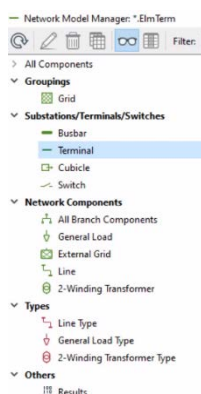


Figure 26. Screenshot of a grid topology modelling software

Consult of the CIM standard provide more information about the classes modelled in the grid topology. The relationship between them can be visualized in Figure 27.

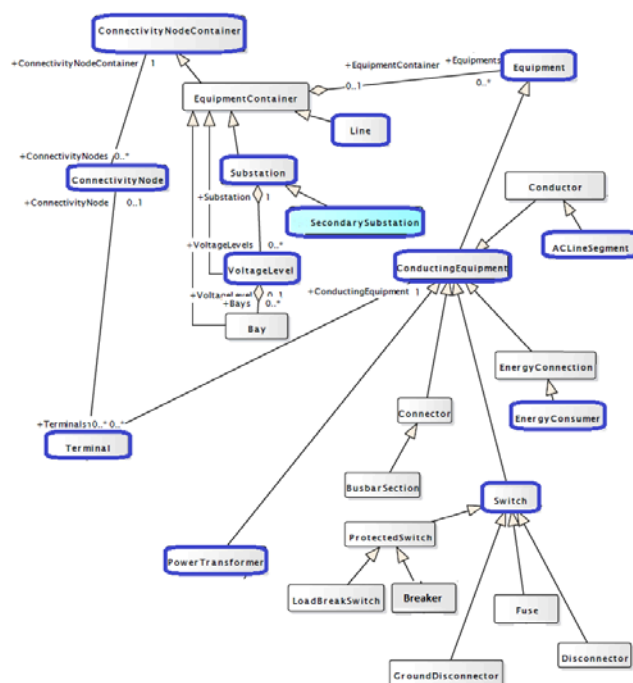


Figure 27. IEC 61970-301 CIM standard classes for representing the electrical view of the distribution grid

DigSilent was used as the reference grid topology modelling software tool, but it is only used in the Spanish demo. To reproduce the model obtained to other demonstrators, it was exported using the .XML files format under the CIM standard, in which the following entities appear:

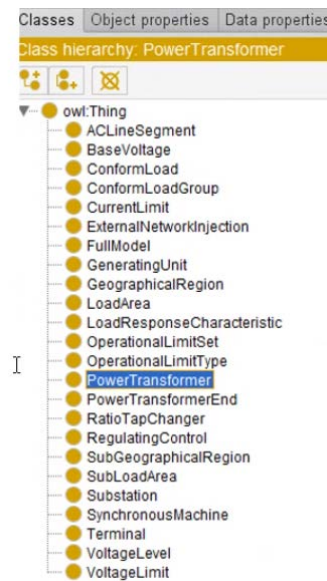


Figure 28. List of objects generated by DigSilent for the Spanish demo using CIM standard

The .xml files generated under the CIM standard are static and only the information considered relevant about them is consulted. The objects listed in Figure 28 correspond to different classes of the CIM standard, which have their own hierarchy, where child classes inherit attributes from their parents, and complement them with specific attributes.

In order to define the FLEXIGRID CIM, it is mandatory to know how to map the CIM standard in the form of JSON files to integrate it with the FUSE platform for all the FLEXIGRID project demo sites, presenting entities to other partners and preventing them from handling the .XML files directly. The names that have been defined for the FLEXIGRID CIM entities in Figure 24 have been based on the list of objects exported under the CIM standard with the DigSilent simulation software shown in previous Figure 28.

In almost all cases, there is a direct relationship between this list of entities in Figure 24 and those listed in Figure 28. However, CIM standard does not provide specific entities for generation and storage, but the rest of the model could be mapped directly. As such mapping exists, they can be handled with the files exported by DigSilent or any other simulation software respecting the CIM standard. If the CIM standard does not support the definition of a new entity, other data models as SAREF or FIWARE will have to be considered to include the entity in the FLEXIGRID CIM.

Translating those entities to JSON prevent other partners from having to handle the generated .XML files directly. Thanks to the work carried out in WP4, it is known what will be presented in FUSE and what will be used directly from the DigSilent simulation developed by CIRCE. After analysing the work in WP4, it was also determined that the information shared by OPC protocol does not need to be represented in the JSON model yet.

An example on the entities' creation employing FLEXIGRID CIM appears in Annex 5.

## 4. T5.3 Cybersecurity requirements, access control and data privacy mechanisms

Over the last couple of years, the burgeoning Internet of Things (IoT) has made it possible to connect anything and everything to the internet. It has led to a digital disruption in the physical world as we know it by changing the way we use technology. The growth of Machine-to-Machine (M2M) communications over the last decade has provided a communication paradigm that has enabled ubiquitous connectivity between devices along with the ability to communicate autonomously without human intervention. The IoT is an interconnection of uniquely identifiable, embedded, computing devices that can transfer data over a network without requiring human-to-human or human-to-computer interaction. As the smart grid evolved, IoT has emerged as an enabling technology to the grid. Since an IoT based smart grid is a complex architecture involving millions of IoT nodes and devices throughout critical power facilities and systems in one network, it represents the single biggest attack surface. As the ubiquity of IoT technology infiltrates further into a smart grid's infrastructure it becomes more and more at risk of cyberattacks. First and foremost, the number of potential attack points across the network is enormously huge, and once a single device is compromised, then the whole grid becomes vulnerable to cyberattacks. Even in instances where the infrastructure is considered relatively secure, but the communications network is not, then the whole system is still at risk. The potential cascade effect of shutting down the electricity grid, make it a key point of cyberattack hence the dire need to protect it at all costs. The National Institute of Standards and Technology (NIST) defines the smart grid as the integration of the last century power grid with the current century development in information and communication technologies. Unlike the traditional power grid, the smart grid maximizes the energy demand distribution, increases efficiency, minimizes losses and makes large-scale renewable energy such as solar and wind deployments a reality. The current grid network is facing severe challenges including recurring blackouts, overloading during peak hours and service disruptions that are never reported in time mainly due to an aging infrastructure. However, the deployment of remote sensing equipment capable of measuring, monitoring, and communicating information about the grid components makes it more connected and smarter. The Smart Grid (SG) is considered as one of the most critical infrastructures and is seen as one of the largest potentials IoT network implementations. Setting up of smart grid networks involves integrating numerous wireless sensors, smart meters, smart appliances, sensors, and other smart objects, all which communicate with each other over a connected network [50]. According to the national institute of standard and technology, a smart grid is composed of seven logical domains: bulk generation, transmission, distribution, customer, markets, service provider, and operations, each of which include both actors and applications. Actors are programs, devices, and systems whereas applications are tasks performed by one actor or more in each domain. Figure 29 shows the conceptual model of smart grid and the interaction of actors from different domains via a secure channel [51].

## Smart Grid Conceptual Model

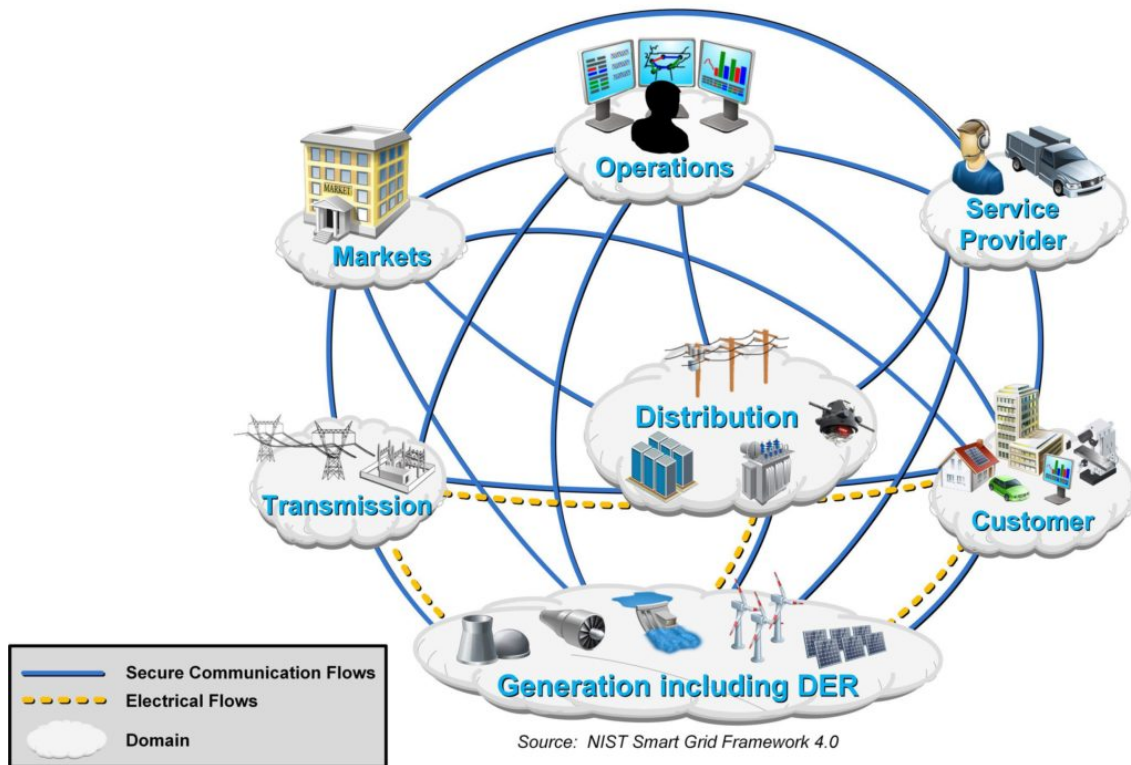


Figure 29. Key elements in Smart Grids

A smart grid system has different pieces such as regional control centres, power generation and distribution units, substations, consumers, tap changers, Information and Communications Technology (ICT) devices, phasor measuring units (PMU), log servers, remote terminal units (RTU), home appliances, protecting relays, Intelligent Electronic Devices (IEDs), human-machine interfaces (HMI), circuit breakers, protocol gateways, and smart meters. Smart grid systems have unique objectives, goals, and features to provide robust communication architecture and reliable power supply. There are some important assets to consider for efficient operations in smart grid applications. Supervisory Control and Data Acquisition (SCADA) provides real-time controlling and monitoring of the electricity distribution network. Distribution management system (DMS) and energy management system (EMS) are subsystems related to SCADA. SCADA enables the standards for controlling, monitoring, and operation of power in industrial processes. Intercepting or tampering the data damages the grid. Control processes can be performed remotely and automatically with RTUs and PLCs. Various technologies such as VPN (Virtual Private Network), IPsec, firewall, user and device authentication, and intrusion detection system (IDS) are used to secure a SCADA network. Also, access logs and distribution control commands are very vital for a SCADA system. Time-tagged data on the network should be synchronized to ensure the reliability and safety of the SCADA system effectively. Advanced Metering Infrastructure (AMI) is the integration of various technologies that provide advanced connections between the control centre and smart meters. The IoT-based smart grid enables that AMI can be implemented easily. AMI is also known as smart metering. HAN, smart meter, operational gateway, and meter data management system are the main components of AMI. AMI is responsible for collecting, analysing, storing, and providing measurement data sent by

smart meters towards authorized parties. Additionally, AMI provides transmitting software updates, commands, requests, and pricing-information from authorized parties to smart meters. IoT-based Smart Grid is the empowered form of conventional power lines with IoT technologies. IoT is one of the enabling concepts and plays a fundamental role in the smart grid. The smart grid is considered as one of the most critical infrastructures and is seen as one of the largest IoT applications. Adopting IoT in the smart grid enables large-scale and bidirectional data flow and connectivity throughout the network infrastructure to manage and monitor the energy grid remotely. Through IoT, smart appliances could be efficiently sensed and managed via the Internet. Plug-in Hybrid Electric Vehicle (PHEV) contributes to reducing carbon emissions and reducing dependence on fossil fuels, thus providing a means to support DERs in smart grid applications. PHEV can run on gasoline and electricity. PHEV batteries can be recharged by users at home or elsewhere. Since most PHEV batteries are designed for rapid discharge, PHEVs can provide electricity power to the grid. The vehicle-to-grid concept can improve reliability and increase the efficiency of the electricity grid. However, the trade-off between benefits and costs is still unclear. Additionally, Distributed Energy Resources (DER), Renewable Energy Sources (RES), and communication technologies are other key factors of the smart grid. Communication across the power line happens through wireless, wire cables, fiber-optic links, microwave channels, and ethernet where a wide range of bandwidths are implemented.

#### 4.1. Cyber-security in smart grid: State of the Art

The main objective of this chapter is to collect the information available from the State of the Art regarding cyber-security in the smart grid context. The main sources of this collection are extracted from recent papers and projects in this field. The focuses of the research and analysis performed refers to the four areas identified into the DoA:

- Equipment security
- Communication security
- Data security
- Platform integration security

The constant advancements in ICT contribute to the development of the traditional electricity grid into the smart grid. However, one of the significant disadvantages of smart grid development is the cyber-security issues that this process implies. Cyber-security concerns slow down the progress of smart grid applications. Nevertheless, steady improvements will enhance the smart grid experiences in the next years. Smart grid cyber-security issues include ensuring the Confidentiality, Integrity, and Availability (CIA) triad of the control systems and ICT. CIA triad is essential to both communication infrastructures and the protection, operation, and management of energy [52]. There are many additional interrelated requirements to ensure cyber-security in smart grid applications highlighted in the following list:

- Objectives:
  - Confidentiality: the protection of data from unauthorized access or disclosure.
  - Integrity: the prevention of data from unauthorized alteration and destruction.
  - Availability: the protection of the information system from breakdown.
- Requirements:
  - Authentication: together with identification are the key processes of confirming the identity of a user or device to defend the smart grid system from unauthorized access.
  - Authenticity: is necessary to verify that transmitted data was received exactly as it was sent and the parties involved are who they claim to be.
  - Authorization: refers to blocking access to the system by unauthorized people or systems without permission.
  - Accountability: together with auditing, provides that smart grid can be traceable and recordable.
  - Privacy: requires that consumer data cannot be obtained by unrelated people and used for different purposes without customer's permission and can merely be utilized for defined permissions.
  - Dependability: the capability of a system to achieve its services in timely and accurately manner, by avoiding common and serious internal faults.
  - Survivability: the ability of a system to perform its task and thus preventing malicious, intentional or unintentional faults on time.
  - Safety Criticality: refers to systems that can potentially lead to severe outcomes due to the existence of some unexpected situations such as earthquakes, floods, tsunamis which may result in substantial physical damage, human injuries, or even deaths.

The analysis performed on the State of the Art of the cyber-security for Electrical Power and Energy Systems takes advantage of the ongoing work under the EU SU-DS04-2018 call. The work reported in the deliverables of the quoted projects has been analyzed to further characterize the cyber-security context of FLEXIGRID. Since all these projects are still running, not all their expected outcomes were ready to be exploited within the security and design tasks of FLEXIGRID. Within the overall projects of the SU-DS04-2018 call (CyberSEAS [53], ELECTRON [54], EnergyShield [55], PHOENIX [56], SDN-microSENSE [57]), the most profitable interactions have been performed with the EnergyShield consortium. In particular, the cyber-security culture framework developed in EnergyShield has been examined in conjunction with the European partners in charge of its development to ease the security requirement process of the FLEXIGRID platform, the connected services, and the pilots. Starting from the knowledge modelling defined within the quoted framework, the work performed in Task 5.3 (Cybersecurity requirements, Access Control and Data Privacy mechanisms) led to the exploitation of the characterization of the most relevant domains that could be applied to the FLEXIGRID scenarios. Since the target of the EnergyShield framework is designed to be easily mapped on single companies, the exploitation of such tool in a distributed scenario with multiple stakeholders that cooperate to reach a common goal, such as the one conceived by FLEXIGRID, led to the necessity of a dedicated approach. As a result, the main outcomes exploited from such a project refer to the identification and characterization of the most relevant parameters in the sub-domains selected to properly classify cyber-security approaches. Figure 30 highlights the levels, domains and dimensions foreseen by the EnergyShield culture model.

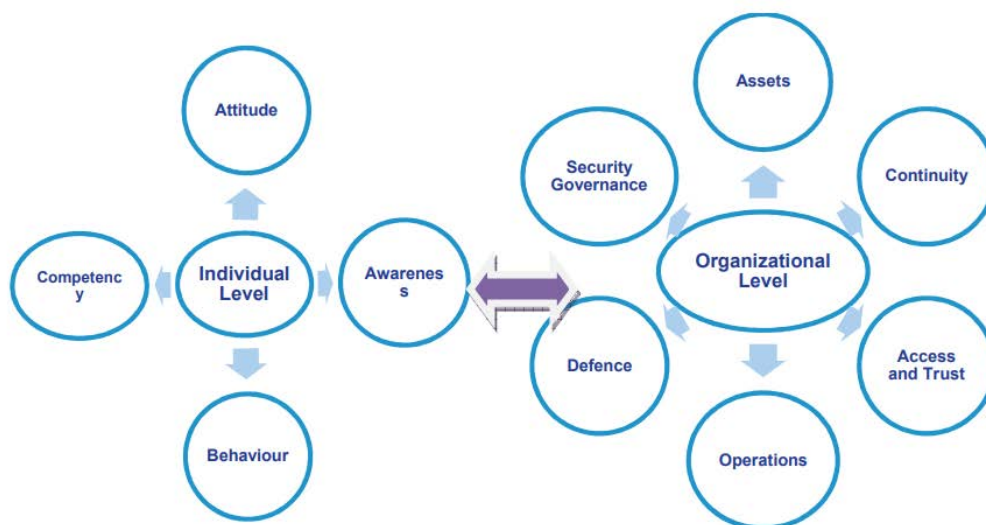


Figure 30. EnergyShield levels and dimensions

The security activities performed in FLEXIGRID focused on the Organizational Level. The Individual Level, referring to each employee's cyber-security culture, was considered out of the scope of the analysis foreseen by the project. Since the solutions built are enabled through a distributed set of components located in three main layers: the pilots, the cloud platform and the remote services, the domain analysis has been performed and adapted coherently. The information characterized and modelled here is then properly addressed in the design and implementation chapters through the analysis and activities performed and detailed.



Figure 31. EnergyShield Asset dimension and domains

The Assets dimension (Figure 31) includes the designing, development, documentation and implementation of security policies and procedures that aim to protect an organization by enforcing several levels of confidentiality, availability, and integrity controls. Within the Asset dimension, the most relevant domains considered in the project are:

- Network configuration management: Establishment, implementation, and active management of the security configuration of network infrastructure devices using a rigorous configuration management and change control process to prevent attackers from exploiting vulnerable services and settings.
- Network infrastructure management: Management of the ongoing operational use of ports, protocols, and services on networked devices to minimise windows of vulnerability available to attackers.
- Personnel security: Management of the proper authentication and authorization level controlling personnel and visitors' access in the physical facilities of the organization.
- Physical Safety and Security: Establishment, implementation, and active management of facilities' physical security.



Figure 32. EnergyShield Continuity dimension and domains

The Continuity dimension (Figure 32) includes the planning, development, documentation and implementation of the security policies and procedures that aim to ensure operations, services and production continuity for an organization while safeguarding the reputation and interests of key stakeholders in case of disruptive incidents. Within the Continuity dimension, the most relevant domains considered in the project are:

- Backup mechanisms: The backup procedures that are in place to avoid loss of critical information and provide a level of acceptable business continuity in case of incidents.
- Continuous Vulnerability Management: Continuous acquisition, assessment, and elaboration of new information to identify vulnerabilities, remediate, and minimise the opportunity window for attackers.



Figure 33. EnergyShield Access and Trust dimension and domains

The Access and Trust dimension (Figure 33) includes the design, development, documentation and implementation of business procedures that aim to ensure appropriate access to resources across the organization while clarifying different roles and permissions. Within the Access and Trust dimension, the most relevant domains considered in the project are:

- Access Management: The processes and tools used to track/control/prevent/correct secure access to critical access according to the formal determination of which persons, computers, and applications have a need and right to access these critical assets based on an approved classification.
- Account Management: Active management of the life cycle of system and application accounts. Their creation, use, dormancy, and deletion, minimise opportunities for attackers to leverage them.
- Communication: Various controls aiming at protecting data, information, and systems during communication procedures.
- Privileged Account Management: The processes and tools used to track/control/prevent/correct the use, assignment, and configuration of administrative privileges on computers, networks, and applications.
- Role Segregation: The proper appointment of roles and responsibilities ensures their segregation in various processes and procedures, to avoid possible issues such as conflict of interests.



Figure 34. EnergyShield Operation dimension and domains

The Operations dimension (Figure 34) refers to the administration of business practices to create the highest level of efficiency possible within an organization while considering the security aspects that safeguard its results. Within the Operations dimension, the most relevant domains considered in the project are:

- Efficient distinction of Development, Testing and Operational Environments: Proper segregation of the development, testing, and operational environments.
- Risk Assessment: Conducting a risk assessment to find any vulnerabilities in the organisation repeated at regular intervals or when significant changes occur.



Figure 35. EnergyShield Defence dimension and domains

The Defence dimension (Figure 35) focuses on the foresight to have planned, acquired, and properly configured all technical assets necessary for the improvement and efficient operation of its information security. Within the Defence dimension, the most relevant domains considered in the project are:

- **Boundary Defence:** Detection/prevention/correction of the information flow transferring across networks of different trust levels with a focus on security-damaging data.

**Cryptography:** All the cryptographic controls used by the organization.



*Figure 36. EnergyShield Security Governance dimension and domains*

The Security Governance dimension (Figure 36) focuses on the design, development, documentation, and implementation of policies to effectively plan, manage, and improve an organization's information security. Within the Security Governance dimension, the most relevant domains considered in the project are:

- **Audit Logs Management:** Collection, management, and analysis of event logs that could assist in detecting, understanding, or recovering from attacks.
- **Incident Response and Management:** Protection of the organisation's information, as well as its reputation, by developing and implementing an incident response infrastructure.

Annex 6 recaps further details on the cybersecurity in the 4 areas of interest in a smart grid.

## 4.2. Risks and threats analysis

In order to characterize the risks and threats within the FLEXIGRID scenarios, it is important to analyse and distinguish them from the known cyber-attacks. The model of the threats started with the STRIDE analysis, valid for any generic software system. STRIDE is an acronym for six threat categories: Spoofing identity, Tampering with data, Repudiation threats, Information disclosure, Denial of service and Elevation of privileges. This modelling provides security teams with a practical framework for dealing with a threat. It can suggest what defences to include, the likely attacker's profile, likely attack vectors and the assets attackers want most. It can help find threats, rank which is most serious, schedule fixes and develop plans to secure IT resources. Each threat highlighted by the model defines the desired property that the system has to offer to limit that threat. As a result, the main generic property identified are Authenticity, Integrity, Confidentiality, Authorization, Availability and Non-reputability. Thanks to the exploitation of secure protocols (i.e., HTTPS), the CIA triad (Authenticity, Integrity, Confidentiality), is mostly covered by design within the information flows conceived in FLEXIGRID.

Authorization functionalities have been integrated through the exploitation of OAuth2 and Open-ID connect as detailed in the next chapters. The impact of DoS attacks has been reduced thanks to the decoupling approach and the non-reputability is granted by the central role covered by the central platform deployed represented by FUSE. Successively, the risks analysis took advantage of more specific classification on the Smart Grid context.

The current chapter reports the latest classification of attacks in Smart Grid scenarios at different levels. Cyber-attacks can destroy a utility's physical systems, render them inoperable, hand over control of those systems to an outside entity or jeopardize the privacy of employees and customer data. Most attacks usually take one or a combination of four main types of attacks: device attack, data attack, privacy attack, and network availability attack [50].

- **Device Attack:** A device attack aims to compromise and control a grid network device. It is often the initial step of a major attack where one compromised device is used as an entry point, to launch further attacks and compromise the rest of the smart grid network. For example, a compromised sensor might be used to send a virus disguised as genuine sensing data hence spreading it to the rest of the network and infecting the whole grid network. As a cyber-physical system, the IoT-based SG with its millions of devices is at great risk since if one device in the network is compromised, the whole network becomes vulnerable. This is especially the case in Trojan horse attacks on the network. Also, due to the high number of devices in an IoT-based smart grid, auditing the network devices to detect any compromised device is both time-consuming and untenable. Strict access control and authentication measures need to be affected to guard against device attacks.
- **Data Attack:** A data attack attempts to illegally insert, alter, or delete data or control commands in the communication network traffic to mislead the smart grid to make wrong decisions/actions. Since an IoT-based SG is founded on the premise of bidirectional exchange of data between the network devices and the utility, any compromise on the data integrity jeopardizes the justification of an IoT-based SG. A commonly observed data attack is when a customer manipulates the smart meter in order to alter his/her consumption data to reflect lower amounts in his/her electricity bill. Sufficient intrusion detection mechanisms must therefore be employed to ensure that the authenticity and integrity of smart grid data are protected.
- **Privacy Attack:** A privacy attack aims to learn about a users' private or personal information by analysing information from their smart grid network resources. Such information might include electricity consumption data where low or no usage of electricity during certain time periods might be used to infer that the location is most probably not occupied. Using such information, the perpetrator might plan physical attacks like burglary as no one is around. Personal information like credit card information shared with the utility provider might also be targeted in a privacy attack. An IoT-based smart grid contains millions of linked user accounts which might be at risk in a privacy attack. In this era of identity theft, users' privacy and confidentiality must be guaranteed. Personal information must therefore be protected from unauthorized access.
- **Network Attack:** A network availability attack mainly takes place in the form of denial of service (DoS). Its intention is to use up or overwhelm the communication and computational resources of the smart grid network, resulting in failure or delay of

communications. An example of a network availability attack is when an attacker floods a smart grid processing centre with false information that it spends so much time verifying the authenticity of the information at the expense of legitimate network traffic. The centre is therefore overwhelmed and not able to timely respond to legitimate thereby causing delay or failure in communications or outright network outage. Network communication in the smart grid is time critical, as a delay of a few seconds has the potential to impact on the control of grid elements resulting in irreversible damage to both the economy and security of a region. A network availability attack must therefore be handled effectively. A network attack on an IoT based SG might render millions of devices to be offline rendering the SG inoperable as the devices would be inaccessible.

As a result, the *Risk Assessment* domain, defined in chapter 2, is performed both on the cloud and by each company managing the pilots conceived by the project through the information gathered and exposed within the current deliverable and the latter documents produced within the project. The following figure highlights the classification of Smart Grid Cyber-Attacks according to the CIA objective triad [52].

Table 2: Classification of Smart Grid Cyber-Attacks according to The CIA triad

Cyber-Security Objective	Attack Type
<b>Confidentiality</b>	Social Engineering, Eavesdropping, Traffic Analysis, Unauthorized Access, Password Pilfering, MITM, Sniffing, Replay, Masquerading, Data Injection Attacks
<b>Integrity</b>	Tampering, Replay, Wormhole, False Data Injection, Spoofing, Data Modification, MITM, Time Synchronization, Masquerading, Load-Drop Attacks
<b>Availability</b>	Jamming, Wormhole, Denial of Service, LDos (Low-rate Dos), Buffer Overflow, Teardrop, Smurf, Puppet, Time Synchronization, Masquerading, MITM, Spoofing Attacks

To better appreciate the dangers posed by cyber-attacks on critical infrastructure, this section reports a list of the most high-profile examples of cyber-attacks known in literature.

- **Tram Hack Lodz, Poland:** In 2008 a tram system hack in the city of Lodz, Poland escalated to the point where a dozen passengers were injured, making this the first cyber-kinetic attack to result in human injury.
- **Texas Power Company Hack:** In 2009 an employee that had just been fired from the Texas Power Company hacked their network to cripple power forecasting systems. He used his logins that were yet to be disabled.
- **Stuxnet Attack on Iranian Nuclear Power Facility:** In 2009, a worm allegedly created by U.S. and Israeli governments targeting Iranian uranium enrichment devices is believed to be responsible for causing substantial damage to Iran's nuclear program by destroying uranium enrichment centrifuges at an Iranian nuclear facility. *Stuxnet* is a

malicious computer worm that targets SCADA systems by targeting programmable logic controllers (PLCs), which allow the automation of electromechanical processes.

- Houston, Texas, Water Distribution System Attack: In November 2011, the Water Distribution System at the Water and Sewer Department for the City of South Houston, Texas was hacked.
- Bowman Avenue Dam Cyberattack: In 2013, the Bowman Avenue Dam in New York was breached, and the hackers managed to gain control of the floodgates. Investigations showed they could easily have changed the settings related to water flow or even changed the amount of chemicals used in water treatment to catastrophic effect. This would have led to devastating consequences.
- Ukraine Power Grid Hacking: In December 2015, hackers managed to seize control of Ukraine's power grid's connected control system by successfully hacking the grid's supervisory control and data acquisition (SCADA) system using the *BlackEnergy* malware. This caused a massive blackout that left over 700,000 people without electricity for several hours.
- Dyn DDoS Attack: In October 2016, Dyn – an internet service provider, suffered a cyber-attack that disrupted access to popular websites and shut down massive portions of the internet in the United States. The hackers executed a distributed denial of service attack (DDoS). The DDoS attack exploited a system known as the *Mirai* botnet, which scans the web for poorly secured IoT devices that still have factory default usernames and passwords. They then commandeered many insecure IoT devices to request for services from Dyn servers. This fake traffic overwhelmed it causing the site to break. This attack succeeded largely because an astonishingly large number of people don't change default logins on their devices. Since Dyn is one of the entities that route web traffic, its going down caused many websites to be unavailable for a day. Popular websites such as *Twitter*, *Netflix*, *Spotify*, *Reddit* and *SoundCloud* were among those that were affected.
- Ransomware Attack on San Francisco Light Rail System: In November 2016 the light-rail system of the San Francisco city in the US was the subject of a ransomware attack in yet another cyber incident. Quite recently, a company that makes digital teddy bears had its online database hacked and millions of private messages between parents and their children exposed. Most of these devices collect personal information like users' names and telephone numbers, while others such as smart meters can monitor user activities (e.g., when users are in their houses). All these events show how easily hackers can use household or office IoT devices to spy on unsuspecting users.
- *Kemuri* Water Company Hack, US: In 2016, hackers infiltrated the *Kemuri* Water Company's water utility's control system and changed the levels of chemicals being used to treat tap water by manipulating the valves controlling the flow of chemicals.
- Smart Building Attack in Lappeenranta, Finland: In 2016, a targeted DDoS attack shut down heat and hot water in two apartment buildings in Finland in the middle of Finnish winter.
- UK Electric Grid Cyberattack: In July 2017, an electricity grid that supplies electricity in UK and Ireland was attacked. The cyber-attack was targeted at infiltrating the power control systems, in order to enable them to take offline all or part of the electricity grid. It was carried out using some fake emails targeting some senior employees at the power company. The emails contained technical information about the grid network intended to pass them off as genuine mail but were intended to illicit information or make the

users click on links to trigger malicious software in what is known as a spear phishing attack.

- Cyberattack against Saudi Arabia petrochemical plant: An unsuccessful cyberattack against a petrochemical plant in Saudi Arabia in August 2017 was intended to not only sabotage the plant's operations but also cause an explosion that could have killed people. Reportedly, an error in the computer code used by the attackers prevented the explosion from occurring.
- DDoS Attack on Sweden Transport Network: In October 2017, DDoS attacks against the transport network in Sweden caused train delays and disrupted travel services.

### *Impact on costs*

BRIDGE [61] is a European Commission initiative which unites Horizon 2020 Smart Grid, Energy Storage, Islands, and Digitalization Projects to foster continuous knowledge sharing and allow them to deliver conclusions and recommendations about the exploitation of project results as a single voice. BRIDGE EU projects have confirmed that cybersecurity investments are usually the result of a risk management process. Hence, to overcome this barrier, guidance and recommendations should support decision makers to assess underlying risks and estimate the costs and efforts required to implement the respective mitigation activities. The process of information security management and risk management includes assigning priority to risks, establishing a budget for the measures to be implemented, and finally implementing and maintaining the selected risk reduction measures (i.e., safeguards). In this respect, it is important to previously identify the current security mechanisms and evaluate their effectiveness.

Considering and choosing the appropriate measure requires a cost/benefit analysis approach. Since a clear characterization of the costs introduced by the risks exposed in the previous section highly depends on the type of attack, the involved hardware and service offered and, on specific IT details of the companies that manage the different pilots, a first global overview has been performed on known significant attacks around the world.

The following picture (Figure 37), produced by *Specops Software* and reported by *Visual Capitalist* [62], report the most significant attacks from 2006 to 2020 divided by country exposing relevant economic impacts of cyber-crime costs.

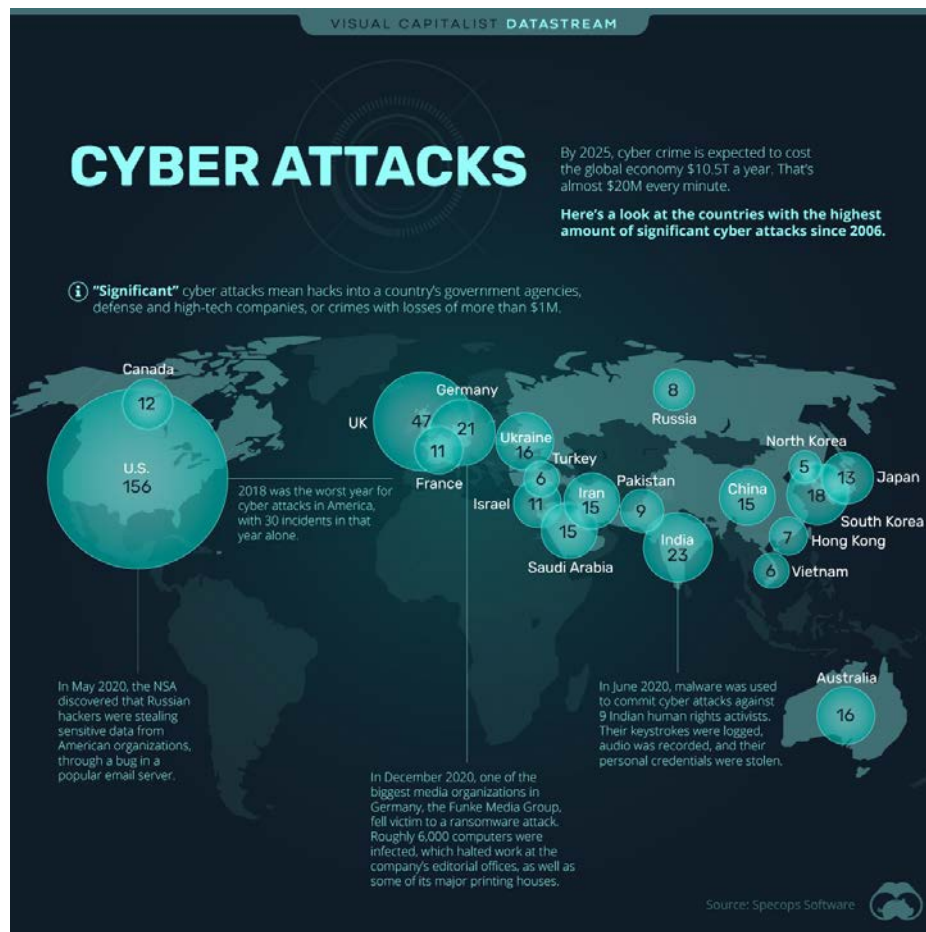


Figure 37.- Most significant cyber-attacks from 2006 to 2020

The costs for the global economy and the prediction exposed clearly highlight the necessity of increased regard of cyber-security aspects. Furthermore, the possible dangers exposed by the examples reported in the risk analysis should lead to a significant investment in the security features applied by companies, their hardware, and services. In FLEXIGRID, the costs introduced by an interruption of all the services managed by each pilot company should be further investigated and afterwards determined since it strongly depends on the condition of the interruption and the DSO. Specific costs of hardware devices replacement due to misbehaviour commands is determined by the kind of damage produced, the eventual condition of the destruction, and the actual number of devices affected.

Typically, in terms of financial losses, there is no current contract (e.g., flexibility) that if any of the developed services stopped could create a loss. However, the benefit of the provided services still needs to be trialed (i.e., the peak shaving service and the energy cost optimization service for the Greek Pilot) to understand better the loss of savings. As an example, the cost to replace the full hardware utilized in the Greek demo (i.e., PV systems, battery systems, EV charging points, smart meters, and energy analysers) could be estimated at about 200k EUR.

The possibility to inject harmful instructions into the hardware present in the FLEXIGRID pilot is prevented by the controllers embedded in each machine that locally validate the values received on their interfaces. Furthermore, the economic impact of the risks introduced by the FLEXIGRID services on the pilots has been reduced through the decoupling control approach provided by

design. Since the possibility to enable hardware control through remote algorithms results is punctually managed by the companies that own and supervise each pilot, there are no additional risks introduced in these services exploitation. This is possible since the main logic that forward cloud algorithms results is located within the pilots. Both the frequency of actual commands sent to the hardware and their sanitization are demanded by client-based software running on each site. As a result, none of the cloud software produced embeds any pilot-related credentials or has the means to directly interact with the physical machines exploited. Consequently, the proprietary networks, data, and hardware of the companies involved are not exposed in any way to the exploitation of the FLEXIGRID services.

### 4.3. Cyber-security framework design

Starting from the final diagram representing FLEXIGRID's reference logical architecture depicted in D5.2 [2], see Figure 3, where modules are separated into five layers corresponding to each of the SGAM (Smart Grids Architecture Model) interoperability layers, the security design focused on the communication interfaces between each pilot and FUSE, enabled by its adaptors. Successively, the RESTful services offered by the cloud platform have been analysed and defined accordingly to the overall application's needs.

On-site communications security is managed by each pilot owner through their IT and security experts since these take advantage of the private networks in place. Due to the sensitivity of their infrastructures, their networks have not been extended or modified in any way to prevent the introduction of threats to services provided by these companies that do not involve FLEXIGRID. In addition, the *Efficient distinction of Development, Testing and Operational Environments* domain is managed both on the cloud platform and on each pilot scenario through the instantiation of dedicated environments for the different phases of the software life cycle. Furthermore, the possibility to deploy the FLEXIGRID applications within the same cloud platform that is hosting the datasets has been denied to further isolate the overall solution and to prevent eventual software vulnerabilities to affect the central architectural point represented by the FUSE platform. Therefore, each software service built is deployed within the responsible company.

#### *Architecture security design*

Due to the complexity and sensitiveness of the smart-grid scenarios, the design process of the overall information flows and the consequent security framework followed the keep it simple, stupid (KISS) principle [63]. It states that most systems work best if they are kept simple rather than made complicated; therefore, simplicity should be a key goal in design, and unnecessary complexity should be avoided. As a result, the harmonization of the data managed by the FLEXIGRID platform and the available approaches to enable end-to-end communication among the overall parties have been defined as the most simple and secure way to interconnect remote solutions. In particular, the idea behind the architecture design is to avoid any distribution of security credentials that could enable direct or indirect access to pilot networks, their sensitive hardware, and data. This solution is possible by exploiting the FUSE platform as the only cloud server able to transfer measurements from the fields and commands estimated by the software services through punctual requests. The quoted approach allows making equal all the communication approaches among the FLEXIGRID components. Furthermore, it distinctly separates all the software produced, preventing single vulnerabilities to impact the overall chains of a solution building.

Even by following the latest security solution exposed in the state of the art, without applying relevant investments and effort on the software code quality, continuous updates, management, and monitoring of all the solutions applied, the possibility to react late to security breaches and malicious attacks is not neglectable. For example, a single exposed node within a VPN could easily put in danger all the other nodes. As a result of the decoupling mechanism and the central role represented by the cloud platform, both the *Network configuration management* and the *Network infrastructure management* dimensions are overseeded by the security solution developed and deployed within FUSE. Furthermore, all the dimensions quoted in the *Access and Trust* domain, defined in chapter 2, are centrally managed by the cloud platform through the OAuth2 and OpenID-connect standards that clearly match the distributed credentials enabling access to the overall features built. Consequently, the domains referred to the *Access*, the *Accounts*, and the *Role segregation* management, are performed through Keycloak and the Role Based Access Control (RBAC) applied. Finally, *Communication* domain aspects take advantage of the secured protocols selected and the exploitation of the JWT (JSON Web Token) punctually obtained by the involved parties.

#### *Network zones definition*

In the FLEXIGRID platform architecture, we distinguish between two types of areas: the trusted and the untrusted areas. The trusted areas are demarcated by the boundaries of every single package and by the FLEXIGRID platform itself. Security mechanisms are applied to all the communications involving the Public Interfaces. The Security Framework, in charge of providing those features, monitor all the requests sent towards the cloud ecosystem. At this stage, the FUSE backend is the main public access point where external clients can request FLEXIGRID services. The untrusted area comprises external applications and networks exploiting platform services.

The cyber-security design process, by taking into consideration modularity and scalability aspects, enforces the adoption of the same Authentication and Authorization mechanisms in all the interaction foreseen by the FLEXIGRID Platform. All the public APIs of the Platform are protected by the Security Framework selected, OAuth2 compliant, further detailed in the following paragraphs. External applications can be enabled to interact with the platform by exploiting the defined HTTPS endpoints. The public Authentication API provided by the Security Framework allows registered clients to obtain the necessary OAuth2 credentials to interact with the FLEXIGRID Platform. Successively to the reception of the grants, it is possible to exploit the cloud functionalities through the secured endpoints. All the software components interact with the Security Framework and trigger asynchronous communications.

As highlighted in the current document, the Public APIs allow external enabled applications to exploit the platform features. When any application needs to interact with the platform servers, it is mandatory to follow one of the available authentications flows later described, to receive a temporary token for the specific resource of interest. The complexity of the security management of the system providing heterogeneous services and interacting with different actors, could introduce severe vulnerabilities. The solutions required to protect against unforeseen vulnerabilities are arduous and often require the introduction of asymmetric changes in the overall behaviour, leading to intricate systems. As a result, the adoption of a common security strategy by design for the overall software components is a fundamental aspect to protect sensitive resources. As already highlighted by the Information Flows, the data gathered in each pilot, is sent to the cloud public endpoint together with the access tokens

punctually obtained from the security framework. Other important features provided by the platform allow harmonizing the data received and trustfully store the data gathered on the field. Finally, the platform provides ways to retrieve the data, securely gathered and authenticated through the components briefly described above.

In addition to the features applied to the entry access of the platform, the overall architecture design, also considering the security analysis performed, lead to the adoption of approaches and tools further quoted in this chapter. In particular, the first security aspect faced is remote access to the cloud system where the overall components of the platform are deployed. It has been configured to grant only a restricted group of the project developers for debugging purposes and by preventing unauthorized access. Then, all the public and sensitive endpoints exposed, providing ways to interact with the platform itself, are protected by a reverse proxy named Traefik [64]. By exploiting the same tool, it is possible to manage balancing on the load introduced by simultaneous requests. It is configured to redirect only TLS encrypted and CA-certified requests to all the components of the platform in a secure and bounded way. By exploiting the TLS protocol, confidentiality and data integrity are guaranteed in the communication between any client and the platform's public endpoints.

Plain communication between a generic client and the platform has been disabled to prevent eavesdropping from attackers constantly sniffing Internet messages. In order to increase the overall security, all the single software composing the platform is confined inside dedicated containers to limit any possible breach generated by unforeseen vulnerabilities to the least number of resources present on the cloud machine. Another important security feature necessary to protect the system against a vast range of known cyber-attacks, referred to as code injection, is the sanitization of the data provided to the platform. Before the forwarding process towards any database or sensitive system, data is properly parsed and controlled to avoid this class of vulnerabilities.

#### *Authentication, authorization, and accounting solutions*

The Security Framework tool selected for providing both client authentication and authorization features is Keycloak. The quoted tool permits the exploitation of known security mechanisms to protect access to exposed resources by providing run-time Identity and Access management. The authentication approach offered by Keycloak follows the specification known as OAuth2, an open security standard designed to provide ways to grant access to sensitive resources by exploiting a simple token, without the need of providing any user or password credential to the resource servers.

The main actors conceived by the standard are:

- Authentication Server
- Resource Server
- Client Owner
- Resource Owner.

The necessary tokens to access protected resources are dynamically obtained through HTTPS requests to the authorization server, exploiting previously generated clients on it. The design process led to the configuration of a Keycloak Realm where credentials, roles, and users able to interact with the platform components are defined. The only token type allowed by the FLEXIGRID Platform is the JSON Web Token (JWT). This standard allows the Authentication server to embed, in a single signed message, other than the basic claims defined by the standard,

detailed information about requesting clients such as the assigned realm roles, and scopes. Consequently, resource servers are agnostic with respect to the users and can exploit the information provided on the token itself to securely enable access on each endpoint. Furthermore, the user Identity information is obtained through the OpenID Connect standard via ID tokens. These tokens take the form of a JWT that is signed with the private key of the issuer and can be parsed and verified by the application. Inside the JWT are a set of defined property names that provide information to the client application.

The Keycloak clients and users are allowed to interact with the system, by taking also into account the roles foreseen and the grant level for each specific resource, which are defined accordingly to their known requirements. All the authorized applications embed the client credentials built in a way that allows to securely obtain time-limited tokens valid to exploit platform features.

The design of both the authentication and the authorization system follows the IEEE best practices. In particular, the Authentication as a Filter approach is exploited with Keycloak as a central point to obtain valid tokens necessary for interacting with all the public platform endpoints. Then, each individual resource is designed with a dedicated authorization approach. Role-based access control is applied to all the developed endpoints. This process refers to the Token signature and content verification conceived by the OAuth2 standard. Furthermore, Centralized authorization features are provided for fine-grained controls over requesters permissions towards specific resources hosted by server components.

The Protection API defined by the OAuth2 standard allows to enforce decisions through permission analysis applied on specific resources with a vast range of Policy. After the definition of the Information Flows, the interfaces between the cloud components have been finalized and the consequent roles have been applied to each Keycloak client. To enforce modularity, security, and to provide isolation in all the single segments of the platform communications, distinct roles and scopes have been defined. The resource servers quoted act as interfaces and provide ways to request specific functionalities from the group of components deployed in each package.

The overall software components composing the platform, by exploiting communication over the internal docker networks only, are isolated and secured by design. Each secured endpoint is mapped to a Keycloak resource and configured with dedicated permissions. The reason behind this is to allow specific isolated settings for clients with different levels of trustiness and different visibility on the overall resources exposed by the FLEXIGRID Platform. The isolation between the permissions applied on the overall functionalities allows also to reconfigure security of every single link in case of identified security breach without affecting the interfaces still safe. Furthermore, future functionalities exposed by a package can reuse the same security settings applied by other endpoints with the same level of grant. The list below describes the available endpoints exploited by external applications to access Platform resources:

- Obtain Access Token: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/token>, depending on the authentication flow exploited the necessary parameters will change. The mandatory ones, used in the client credentials mode are Client ID, Client Secret and Grant Type.

- Obtain Temporary Authorization Code in Token: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/auth>, dedicated endpoint for Authorization code flow.
- Refresh Token: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/refresh>, it enables the possibility to obtain a new access token within the same keycloak session.
- Obtain ID Token: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/token>, it is necessary to add the OpenID scope in the request. Resulting JWT can embed mandatory claims for specific user attributes.
- Get User Info: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/userinfo>, returns the user attributes.
- Get Public Certificate of the Realm: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/certs>, returns the public key used on the specified realm. It enables the possibility to verify the signature of the JWT dynamically without the need of embedding the keys within the resource servers.

Keycloak provides endpoints with different functionalities to simplify the management of the security parameters required to interact with the FLEXIGRID Platform. Any trusted server can exploit these resources to boot with the latest permissions set through the Authentication Server GUI (Graphic User Interface). The list below describes these resources, dedicated to trusted servers.

- Get List of resources id of the specified Realm: [https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/authz/protection/resource\\_set](https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/authz/protection/resource_set)
- Get Permissions details of a specific resource: [https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/authz/protection/resource\\_set/{id}](https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/authz/protection/resource_set/{id})
- Get Introspection analysis on Token: <https://unified-api.fuse.flexigrid-h2020.eu/keycloak/auth/realms/FLEXIGRID/protocol/openid-connect/token/introspect>

In addition, if the resource server implements a worker that periodically triggers those requests and adjusts these settings, any security update performed on Keycloak will be automatically reconfigured in all the servers of interest. The resource servers composing the FLEXIGRID platform provide different functionalities through the set of security approaches briefly quoted before. The services built, allow the introduction of middleware functionalities in front of the servers' endpoints. In addition, authenticated clients and communication flows guidelines are provided.

Trusted developers can access the resources of the platform through the Direct Grant Authorization Flow by exploiting a dedicated user, built with these purposes for each application, together with a Keycloak public client. Trusted applications developed in WP4, can access the cloud platform without leading to any login process. These applications securely embed a Keycloak client with restricted access to the Platform resources. In particular, the OAuth2 authentication flow exploited to obtain access tokens is a client credential.

The security framework is also ready to support all the other flows foreseen by the standard, for future exploitation of the platform features. The list of authentication flows supported, and their usage guidelines is here provided:

- Client credentials: It is designed for non-interactive machine-to-machine communications.
- Implicit: It is mainly designed for single-page JavaScript applications. It also requires an authorized username and password.
- Resource Owner Password Credentials: It is designed for highly trusted applications (e.g., first-party application), as it directly handles user credentials. It also requires an authorized username and password.
- Authorization Code: It is designed for applications that must be able to interact with a user agent (e.g., a browser) in the environment. It also requires an authorized username and password.
- Authorization Code with Proof Key for Code Exchange (PKCE): It is designed for mobile applications exploiting specific code challenges and verifiers. It also requires an authorized username and password.

The OAuth2 standard enables the possibility to exploit refresh tokens, within a specific time window, to obtain new access tokens without repeating the authentication flow. For security reasons, the refresh request must provide the same client security credentials given in the first flow. This approach allows maintaining an authentication session open with Keycloak. The Security Framework allows us to identify the active sessions and revoke the validity of the clients and their tokens. The already delivered tokens will still contain valid credentials embedded and will still be accepted by the Signature Token Verification performed by the distributed Resource Servers of the platform, until their expiration. Whenever sessions are revoked, the refresh mechanism is disabled.

In most scenarios, long-term sessions are safely exploitable only by performing Token Introspection Verification on each request. The architectural analysis and the Information Flows show that most of the clients' interactions with the platform imply frequent messages. Coherently with the use cases defined, the Security Framework is configured to allow long-term sessions and to provide only JWT with limited-Term validity. The reason behind this is to avoid an excessive amount of token requests for each data stream conceived.

All the exchanged messages include an access token, obtained through one of the flows mentioned. To further limit the known cyber-attacks, the offline access functionality of the JWT provided by OAuth2 is disabled for all the involved actors.

The decisions described above reduce the necessity of the Token Introspection functionality on resource servers and the consequent additional load on the Security Framework. The possibility to enable Long-Term validity tokens, if necessary (e.g., for scalability purposes), is foreseen only between trusted cloud components. The management and the distribution of the security credentials are fundamental aspects of a platform that involves different applications and consequently different stakeholders. At this stage, the enabled apps embed clients and secrets built during the project. The possibility to exploit OpenID Connect functionalities, enabled by the Security Framework, to enforce the security of user-dedicated resources has been considered. This option is enabled for future extensions and different usages of the platform features.

The Security Framework can be also federated with other Identity Providers to simplify the login process through the integration with other systems and the exploitation of the ID Tokens. Other than Authentication and Authorization features a common necessity for public platforms is to provide accounting mechanisms. This feature is provided for administrators, by the internal component of the cloud platform. Accounting refers to the ability to measure and logging the resources that a user or software consumes during the access and exploitation of platform features. This last aspect is essential to provide administrators with a way to identify and eventually react to malicious usage and attacks. Any Intrusion Detection System (IDS), focused on anomaly detections on the hosts resources where a platform is deployed, requires constant analysis of the accounting logs generated at run-time.

#### *System actors and privileges*

Starting from the STRIDE analysis performed within the project, the list of tests reported in the following sections will match the security properties required to address the highlighted risks. The final design of the security infrastructure developed within the project is reported in the current deliverable. By exploiting the TLS protocol, confidentiality and data integrity are guaranteed in the communication between any client and the platform's public endpoints. Both the authentication and authorization properties are enabled by Keycloak, the security framework tool selected, that follows the specification known as OAuth2 and OpenID connect. Finally, the availability property of a cloud system is strongly influenced by the number of granted users foreseen by each scenario and involves many different aspects of both hardware and software. Starting from the fixed hardware resources enabled on the cloud machine selected, the availability of the platform services is firstly managed by the Traefik reverse proxy. In addition, a set of controls embedded in the security infrastructure verify the client grants and deny cloud resources usage to limit as much as possible DoS attacks effects.

Furthermore, the deployed mechanisms provide ways to avoid cloud resources starvation by dynamically limiting the exploitation of the API when overloaded. The Retry-After header foreseen by HTTP can be exploited to indicate how long the client application should wait before making a follow-up request. This approach could prevent cloud crashes and could provide useful information to the requester application to synchronize the machine-to-machine communication dynamically and autonomously based on the real-time available resources.

The security tests performed focused on the authentication, authorization, and availability properties since both confidentiality and data integrity are already guaranteed by the exploitation of the encryption mechanisms embedded in the TLS-based communications. In particular, the tests performed to demonstrate the different API visibility enabled through the exploitation of the dynamic JWT, compliant with OAuth2 and OpenID-connect, obtained by the different actors foreseen by FLEXIGRID. OAuth2 is an open security standard designed to provide ways to grant access over sensitive resources by exploiting a simple token, without the need of providing any user or password credential to the resource servers.

As already reported, the Security Framework tool selected providing both client authentication and authorization features is Keycloak. The necessary tokens to access protected resources are dynamically obtained via HTTPS requests to Keycloak, exploiting previously generated clients on it. The design process led to the configuration of a Keycloak Realm where credentials, roles, and users able to interact with the platform components are defined. The only token type allowed by the FLEXIGRID Platform is the JWT. This standard allows the Authentication server to embed,

in a single signed message, other than the basic claims defined by the standard, detailed information about requesting clients such as the assigned realm roles, and scopes. Consequently, resource servers are agnostic respect to the users and can exploit the information provided on the token itself to securely enable access on each endpoint. Furthermore, the user Identity information are obtained through the OpenID Connect standard via ID tokens. These tokens take the form of a JWT that is signed with the private key of the issuer and can be parsed and verified by the application. Inside the JWT are a set of defined property names that provide information to the client application. The clients, the users and the roles signed by the authorization server are used by the e-Security Infrastructure to apply the proper grants and to determine if the requester is allowed to interact with the specific API.

#### *Countermeasures*

As already described in the previous chapters, the decoupling of pilot hardware and cloud services drastically reduced the possible threats and consequent necessity of countermeasures. The standardized approach used to let communicate the overall components developed concentrates on the points of failure of the infrastructure to the APIs enabled by FUSE on the cloud. Thanks to the exploitation of the OAuth2 standard and its security credentials, distributed among the partners in Out of Band (OOB) manner and dedicated for each pilot purpose, can be punctually managed and revoked through the administration functionalities of the Keycloak tool deployed. At any time, credentials enabled for services or pilot applications can be disabled preventing access and exploitation of the developed functionalities.

In addition, the set of management and monitoring tools used on the cloud, such as Rancher, allows to rapidly verify leakages and misbehaviours. Rancher [65] is a complete software stack for teams adopting containers. It addresses the operational and security challenges of managing multiple Kubernetes clusters, while providing DevOps teams with integrated tools for running containerized workloads. This tool has been used to manage the dockerized platform components and the distributed data storage enabled by Elasticsearch.

Furthermore, the redundant process of data sanitization applied both on the cloud and pilot software prevents unexpected scenarios where malicious users try to inject dangerous commands. Finally, the data logging features offered by each software component provide useful information to manage unexpected errors consequent to the misuse of the APIs built and the eventually dedicated credentials leaked and exploited by remote attackers. As a result, the *Audit Logs Management* and *Incident Response and Management* domains, defined in chapter 2, are satisfied through the features put in place through these logging mechanisms and the credentials management.

#### *Pilot security*

Starting from the characterization of the domains and dimensions listed for the FLEXIGRID scenarios, the final security analysis performed is briefly described for each pilot. The overall information gathered via dedicated questionnaires, reported in the annexes, is mapped to provide an assessment of the measures foreseen to protect the hardware and software resources developed. Both the *Personnel security* and the *Physical Safety and Security* dimensions are granted to each pilot by the responsible company through the set of security policies that prevent access to the physical facilities and the deployed equipment to unauthorized persons. In particular, the possibility to interact with the hardware devices managed is limited to known technicians and employees.

### *Spanish Pilot*

The Spanish Demonstrator is divided in 3 scenarios: Scenario 1, aimed at demonstrating upgraded substations and testing grid automation and control algorithms; and scenarios 2 and 3, focused on demonstrating grid protection, fault location and self-healing algorithms. Annex 2 highlights the deployment view and the protocols foreseen in the Spanish Pilot.

The information gathered is electrical measurements of the Low Voltage Grid, the Medium Voltage Grid, and the Load Grid. The actuation foreseen is about MV Automation, LV supervision and OLTC transformer. The on-site communication of the Viesgo network exploits a dedicated VPN accessible to remote users. These exploit a secured https protocol where the applied encryption guarantees confidentiality and integrity. In addition, the communication foreseen relies on web services and IEC-60870-5-104 protocols. The complete dataset needed is gathered by the *ekor.ccp* (MV RTU) and the data are then transferred to Thinlinc Server located within Viesgo. Finally, Thinlinc exploits an SFTP connection towards the FUSE platform to store relevant data for the KPI calculation. In case of unreachable services or communication issues it will be triggered an alarm on the SCADA-based equipment notifying the specific malfunctions.

Furthermore, redundant data storage is located in every RTU, with limited memory capabilities. All the information collected on-site and remotely within Thinlinc will be available only for specific employees exploiting dedicated credentials. The risks of breaches are mitigated by the local protocol used since the access on the hardware is protected with codes and the communication of the devices is encoded.

Further details of the Spanish pilot are reported in D6.2 [66].

### *Greek Pilot*

The Greek Demonstrator focuses on two scenarios targeting the generation and load forecasting module and the congestion management module. The site consists of a hotel resort in Thasos with a 400kVA substation and several bungalows, three of which are equipped with PV and batteries. The substation load is monitored along with a twin EV charging point which is also installed on the site. Annex 2 contains the deployment view and the protocols foreseen in the Greek Pilot.

The information gathered are PV generation, individual buildings load demand, substation load demand, battery status and State of Charge. The actuation foreseen is about the charging and discharging of the batteries, and the switching of the charging points. The communication network used has a specific DNS for the FLEXIGRID assets only, so Energy Box deployed is pre-configured to access the local network via ethernet. Local communications exploit a secured https protocol where the applied encryption guarantees confidentiality and integrity. Communication is available during trial periods and the developed solutions can manage multiple requests at the same time to grant services availability. The complete dataset needed is gathered locally by the Energy Box from all the devices involved in a 15-minute period basis and then transferred to FUSE via https.

In case of unreachable remote services, the PV generating modules will continue to produce energy without the need for any remote control. The battery systems, when they are in manual mode, requires remote control to be operated. Otherwise, they can be switched to automatic mode during a network issue locally. The automatic mode charges each battery when there is excess solar power produced and discharges it during shortages. The EV charging point relay is a digital relay that can be operated only remotely, so if the last setpoint before network loss was

to turn the relay off, that status will remain until the next setpoint. With respect to the equipment deployed, wrong commands could harm the hardware. There are limits in the batteries for safely charging and discharging them, so their violation will affect the warranty and potentially lead to damages in the hardware. Further details of the Greek pilot are reported in D6.3 [67].

#### *Croatian Pilot*

The Croatian demonstrator focuses on two scenarios targeting the coordination of the distribution network flexibility assets & protections schemes in urban districts, and the Virtual Energy Storage for urban building. The site consists of a MV distribution network that is radial in operating regime but is planned as a meshed network with the possibility of changing the network topology, and a residential apartment equipped with smart metering infrastructure. Annex 2 includes the deployment view and the protocols foreseen in the Croatian Pilot.

The information gathered are divided into dynamic and static. Dynamic information includes individual end-user's load demand, substations load demand, voltages and currents measured at circuits of HV/MV substation. Dynamic data include network topology, switching state of a network, information and locations of devices used for QU regulation, on-load tap changer transformers data, data about distributed generators. The actuation foreseen refers to End-user's flexibility (increase and decrease of electricity consumption), change of the network's topology and change of the operational schedule of the considered devices.

Local communications exploit a secured https protocol where the applied encryption guarantees confidentiality and integrity. Communication is available during trial periods and the developed solutions can manage multiple requests at the same time to grant services availability. The complete dataset is aggregated, stored, and anonymized on-site in a HEP-ODS's database and only the subset important for KPI analyses is transferred to FUSE through its secured API.

Based on the services calculations, actuation commands are sent to the HEP-ODS platform and Hypertech Cloud. Depending on the estimated commands, different actions in networks are taken. In case of unreachable remote services, warning and alarms are triggered. The devices deployed are equipped with local storages with limited memory that provides means to support eventual debugging scenario. The data collected on those redundant storages is accessible only by specific employees with dedicated credentials.

A risk of breach where wrong commands are sent could damage the hardware or they could reduce the lifetime of the equipment, e.g., wrong tripping of relays reduces the number of tripping left. Further details of the Croatian pilot are reported in D6.4 [68].

#### *Italian Pilot*

The Italian demonstrator focuses on two scenarios referring to the dispatching platform for MV generation and the Mountainous valley grid operating in island mode. The site consists of the MV grid connected to the Sarentino primary substation, in South-Tyrol. The deployment view and the protocols foreseen in the Italian Pilot appear in Annex 2.

The information collected on-site are several electrical measurements of the MV grid. Local communications exploit a secured https protocol where the applied encryption guarantees confidentiality and integrity. Data persistency is made on-site through the Storage Grid Controller (SGC), where the information needed by the algorithms developed are stored within local files and SQLite databases.

In addition, SCADA information related to the grid and malfunctions are stored through dedicated software. The data collected on those storages is accessible only by specific employees with dedicated credentials. Since the machines used coexist within the same networks with services for the energy supply and management eventual vulnerabilities could lead to dangerous situations. This cyber security aspect regulated by the European directive NIS (Network and Information Security) and its Italian transposition includes the DSOs in the critical infrastructures. Because of the sensitivity of such deployment, the consortium agreed on performing the trials exploiting only software modules deployed directly within the pilot. During the testing phase, the data is stored on-site and then delivered via SFTP to the FUSE to enable KPI evaluation. Further details of the Italian pilot are reported in D6.5 [69].

#### 4.4. Secure platform implementation details

Scalability and performance are important factors in ICT systems, no matter if they are used in monolithic applications or in applications for fully distributed systems. The FLEXIGRID architecture is adopting a general “design for scalability” approach further described in this chapter.

To provide a clearer view on the scalability aspects foreseen by the FLEXIGRID platform, and the consequent services, it is important to further classify the involved topics and the overall resulting considerations. The aspects considered by the design process are reflected on the networking, the processing, and the storage capabilities of the developed system. The main scalability aspects related to the network load are foreseen in two different ways. Thanks to the selection of a cloud-based machine and to the monitoring of the necessary resources, the steps to overcome an excess of requests can be simply managed by increasing the bandwidth provided to the machine itself, basically, to add resources over that node, usually referred as vertical scalability. This process requires changes on the previously signed contract by selecting a greater set of network resources to be assigned on it, with major fees but granting a constant service. Then, the scalability factor can be further increased by adjusting the Traefik load balancer settings and consequently leading to a parallelized and improved forwarding process of the requests, exploiting different machines behind it, usually referred as horizontal scalability.

As already mentioned, the security design and the development strategy allow to safely spread the security infrastructure and the overall sub-components of the platform on different dedicated machines. By following the same approaches exposed above, the disk size and the consequent amount of data that the platform can persistently store can be also improved. On the processing perspective, those necessities can be obtained, as already mentioned for the network load, by signing a different contract and by requesting more hardware resources, without the necessity of any software reconfiguration. Other than that, thanks to the adoption of proper development strategies, without any change on the source code except for single configuration files, the processing capability can be enhanced by increasing the number of processes to obtain a higher parallelization of the required tasks performed by the overall components.

In addition, all the aspects faced in the current chapter can be applied also to each Docker container built to properly isolate and manage every single software instance. To increase the performance factors, the development of the software components exploits the latest frameworks and languages available. In particular, the main platform components are built with

latest Python versions, FastAPI framework and the latest concurrent code methodologies to provide optimal solutions for large-scale scenarios.

The FLEXIGRID platform has been equipped with backup solutions performed via Kubernetes Persistent Volume Claims (PVC). Hence, all information goes to a central NFS server to satisfy the *Backup mechanism* domain. Finally, through the set of logging features enabled, it is possible to monitor and trigger remediation actions to minimize the opportunity window for attackers as suggested by the *Continuous Vulnerability Management* domain, defined in chapter 2.

As detailed in the previous chapters, the exploitation of OAuth2-based interactions and secured protocols makes it possible to face the *Boundary Defence* and *Cryptography* domain, over the communications conceived by the project over public networks. The overall cloud API are protected through a Policy Enforcement Endpoint that delegates the security checks to a main common service integrated with Keycloak. Thanks to the exploitation of Rancher, the FLEXIGRID platform can manage distributed containerized environments such as Kubernetes to host the overall cloud software components highlighted in the functional architecture.

Due to the specific scenario requirements exposed in the previous chapters, the adapters built and integrated into the platform enabling pilot measurements to be stored within the cloud datasets foresee publish-subscribe protocols such as MQTT and AMQP. Since these protocols do not allow the exploitation of the JWT to securely authenticate and authorize services exploitation, dedicated approaches and credentials have been defined to enable secure access towards the platform and the upload of measurements or relevant datasets used for KPI evaluation. Due to the sensitivity of the quoted demonstrator's hardware and networks, the cloud platform foresees the possibility for authorized employees to manually store relevant data on isolated environments through the exploitation of SFTP.

## 5. T5.4 Interoperability in FLEXIGRID'S ICT platform

The present section includes a description of the platform integrating the different software modules developed during the project to foster the interaction among them with an eye on the assurance of present and future interoperability. Hence, it will include the integration towards the end users but also the simplification of how services will be part of the system to allow replication of services and its extension beyond FLEXIGRID pilots.

### 5.1. Interoperability in the Spanish demonstrator

In relation to the Villabermudo I pilot, although it is not possible to upload operational data in real time, they are being recorded since October and the possibility of packaging data, formatting it and delivering it periodically to FUSE so that ATOS can proceed to work with this data and calculate the expected KPIs could be considered.

This way of working may not be ideal but is the most suitable one once known the limitations in what concerns the connection to the private network of a private distributor. In order to not cause any harm to their daily operations, and upon conducting an evaluation of diverse alternatives (such as setting up a dedicated SFTP server to proceed with periodic exchanges), in the end the recommended way to go implies the use of asynchronous tools where data is periodically stored by the DSO and the partners involved in the trial willing to gather such data and perform processing, KPIs calculations and graphical representation using their specific credentials to retrieve it and do so (see Figure 38).

On the other hand, and for the work requiring non-operational data that involves other trials, the interface with the interested partners is already being done. As these data are not going to the control centre, they can be acted upon.

The amount of data downloaded is still to be confirmed because it will depend on the capacity of the hardware installed. A first estimation indicates a capacity to accumulate up to 2 months of information, however, in certain cases, it is possible to refer back to information exchanged up to 6 months ago.

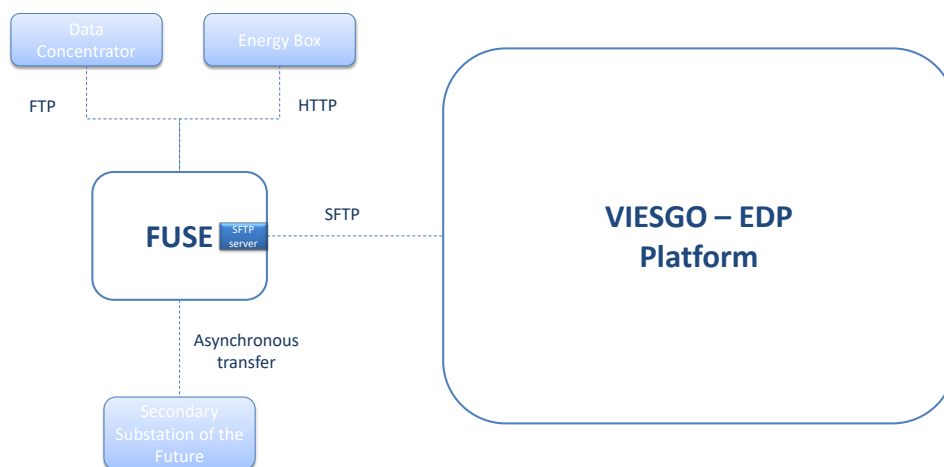


Figure 38. Communications in the Spanish demo site

In relation to the way in which fault information can be extracted from the Entrambasmestas pilot, there is a computer, as well as a buffer of the last 3-4 pulses and when operating events arise (not necessarily faults) it cleans and keeps the most recent ones. It makes it possible to know the distance to the faults and interact between entities involved in this pilot.

Adaptors for Energy Boxes and FUSE platform in both the Spanish and Greek pilots will imply a connection via Hypertext Transfer Protocol (HTTP). Along these lines, and when the moment comes to interact with services developed as part of WP4's job, an Open Platform Communications (OPC) server was set up to facilitate the interaction among modules developed and managed by different partners in the consortium.

The overall way to ingest data in up to four different ways from the demo sites, harmonize it and interact with the database (DB), which is built in Elastic Search, is through the so-called Unified Application Programming Interface (API). Then, a REST (Representational State Transfer) API will be the tool to exchange info from the database with the end users. A dashboard will be implemented to this end, as will be discussed in Section 6.

Such API to facilitate data access and to retrieve data once it is stored in the Elastic Search database, is therefore based on a REST architecture style and supports HTTPS GET protocol for data retrieval.

#### *OPC server in the Spanish Demo Site*

OPC is a client/server technology where one application acts as the server in charge of data provision, and the other application acts as a client using such data. Hence, OPC is an industrial communication standard that enables data exchange between multi-vendor devices and control applications without proprietary restrictions.

One of the main benefits of OPC adoption is the interoperability it facilitates. Suppliers can provide solutions that are truly open, which in turn extends the potential choices in front of the users.

To successfully connect to the OPC server deployed in this context it is mandatory to prepare a Python script that must be accompanied by the corresponding certificates that will be loaded upon connection request.

To fulfil the objectives pursued in this particular case, that implies a bidirectional communication from and to the API, this element requires an update on the methods initially available. This updating works under the following assumptions:

- **POST** methods are used here to upload as many JSON files as needed to the Fuse Elasticsearch Database, by deleting all previous documents before storing the data we ensure that the information saved is always that of the last POST.
- The **GET** methods of each element return the information related to that element, which is, as we have seen, the most recent.
- The "**GET total**" method returns the current information of all the elements.

It is worth mentioning that after valuing them in the first instance, the **PUT** methods, where the values are changed each time data is sent, were finally discarded.

The communications structure between OPC server and unified API is as follows:

- **API to the OPC:** every N minutes, the data stored in FUSE's Elasticsearch database is sent to the OPC server updating the information of its nodes.
- **OPC to the API:** every N minutes the OPC server is checked for updates and the information stored in its nodes is posted to Fuse platform through the Unified API.

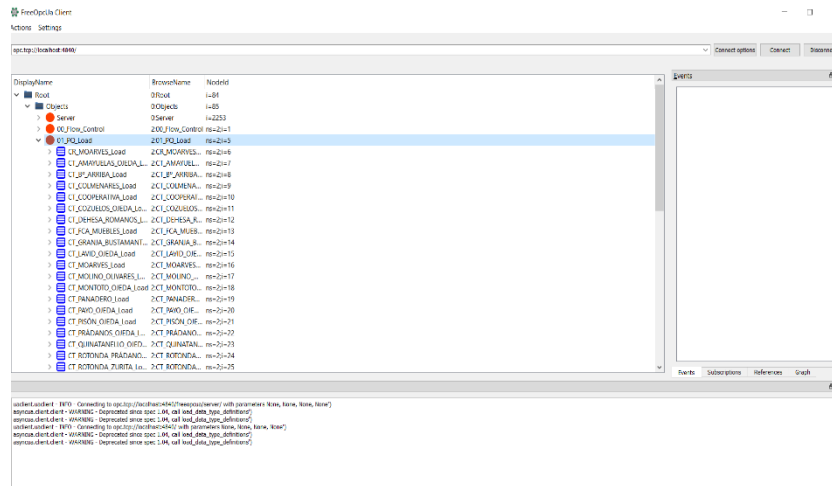


Figure 39. Sample of the OPC GUI showing loads measurements in several nodes

## 5.2. Interoperability in the Greek demonstrator

The vacation resort which hosts the Greek demo site counts on a 400kVA substation and a number of bungalows, three of which are equipped with photovoltaic (PV) tools and batteries. The substation load is monitored along with a double electric vehicle (EV) charging point which is also installed on the site.

The hotel comprises a number of loads that are mainly individual lodge loads, offices and other auxiliary services (e.g., reception building, restaurant, etc.), however, this dedicated 400kVA substation supplies only residential bungalows and the double EV charging point.

Moreover, the hotel is fully equipped with an optical fibre local network (that reaches every bungalow) which is supplied by appropriately hierarchically distributed access points. Each monitored and controllable device (i.e., the inverters, the energy analysers, the energy meters and the control relay for the EV charging point) is networked locally. Their communication with the private cloud storage within FUSE is enabled via the installed Energy Box and appropriate software components/connectors developed in WP5. In addition, the local network is secured, and encryption is employed.

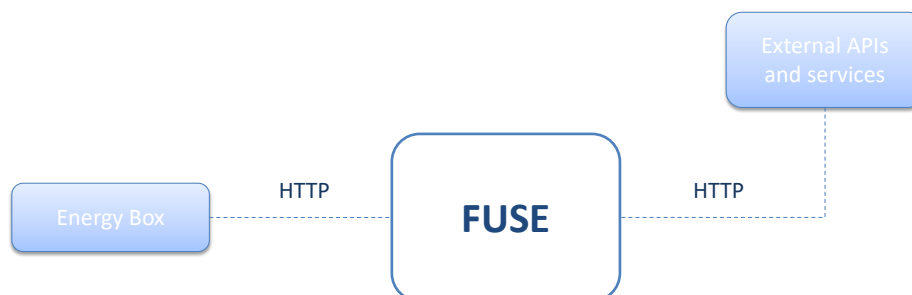


Figure 40. Communication in the Greek demo site

A connection via HTTP will be the base for the adaptor among FUSE platform and CIRCE's Energy Box, when in this scenario information will be exchanged via POST methods.

The algorithms set up in place need to acquire this data sent from the Energy Boxes, specifically figures related to load consumption and power generation, and to do so proceed with a GET method that performs the proper consultations to the API.

Those algorithms require to set up certain endpoints for their setpoints to work fine. An example of this just follows below (see Table 3), in this case referred to *BaseURL/setpoints/battery*:

Table 3 (Code): Example of setpoints in Greek demo site

```
[{
  "Timestamp": "2019-05-02T00:00:00Z",
  "Setpoints": [
    {
      "Bat_min_SOC_220": 0,
      "PF_Set_220H": 0,
      "PF_CTRL_220H": 1,
      "Bat_min_SOC_250": 0,
      "PF_Set_250H": 0,
      "PF_CTRL_250H": 1,
      "Bat_min_SOC_300": 0,
      "PF_Set_300H": 0,
      "PF_CTRL_300H": 1
    }
  ],
  "EV": 1
}]
```

Therefore, and following this approach, the Endpoints for PV Forecast, and Load Forecast can be located in the paths "*BaseURL/forecast/pv*" and "*BaseURL/forecast/load*" respectively.

In the end, the output from these algorithms is made available through requests to the Unified API of FUSE platform for the pilots to extract valuable information, also the calculated KPIs are made available for the perusal of pilots through two different GUIs, namely a Kibana Dashboard for static KPIs and Dash Dashboard for Dynamic KPIs.

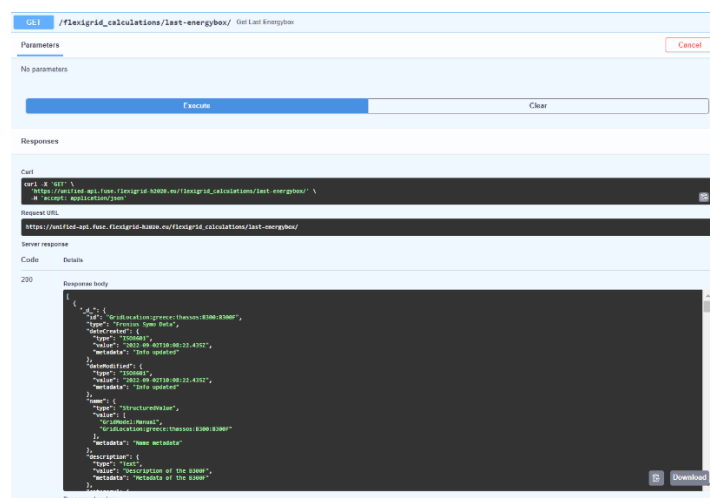


Figure 41. Sample of GET query performed on the Unified API

### 5.3. Interoperability in the Croatian demonstrator

The Croatian pilot location is represented by a single LV customer who is equipped with a smart-meter and is able to establish communication with the HEP-ODS platform and exchange the necessary data through Hypertech cloud and FUSE platform, using AMQP and HTTPS protocols. Calculations made in the HEP-ODS platform are based on the consumption data collected from smart meters and from collected weather data, that are used in necessary predictions. After the calculations, in a case of need, the signals are sent to the end-user's controllable devices. Based on the signals, the controllable devices change the behaviour and help in the network conditions improvement.

Communications in this scenario will be carried out through two different channels, namely a REST API to collect Flexibility forecast and Baseline forecast data via the sending of periodic requests and an AMQP server from which consumption data is going to be continuously fed into FUSE Platform. The picture in Figure 42 reflects the interconnections taking place among the constitutive modules as developed and managed by different entities in the consortium.

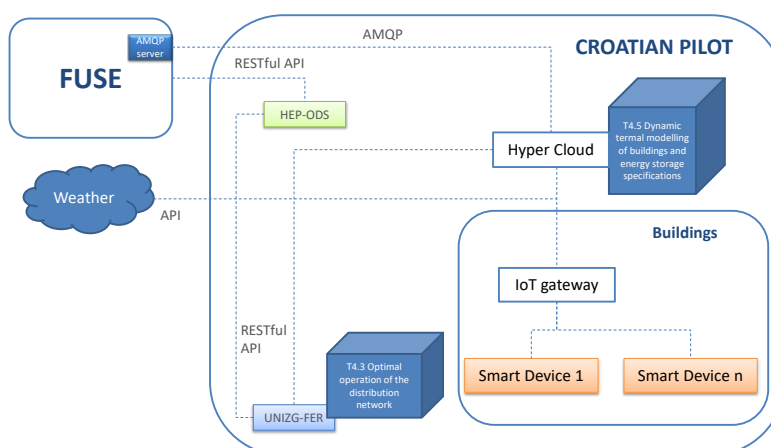


Figure 42. Communications in the Croatian demo site

The REST API is employed to accommodate the flexibility services expected to appear in this context. By sending queries to the Hypertech API comprised between any two start and end points we obtain the mean values for baseline Forecasting and mean down values for Flexibility Forecasting which are then used together with consumption data to calculate the required KPIs and results are shown in our dash GUI. Figure 43 depicts roughly how these interactions take place in this part of the demonstrator site.

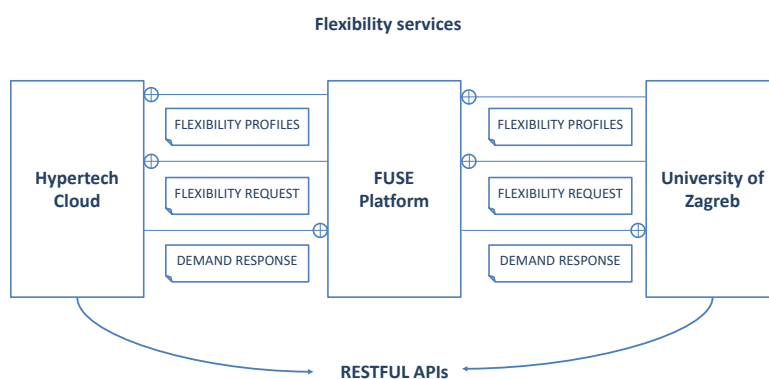


Figure 43. Croatian demo site: RESTFUL APIs for flexibility services

The calls to interact with the API and retrieve the desired information adopt the structure shown below, where these client URLs (cURLs) are the command line tools employed to transfer data to and from the server to the platform:

- Flexibility prosumer: `curl --location --request GET '...'`
- Flexibility per asset: `curl --location --request GET '...'`

One of the most convenient methods to proceed with these kinds of requests implies the use of a tool such as Postman [66], an API platform for developers to the design, build, test and iterate their APIs. Readers may find a screenshot of the GET request in Figure 44.

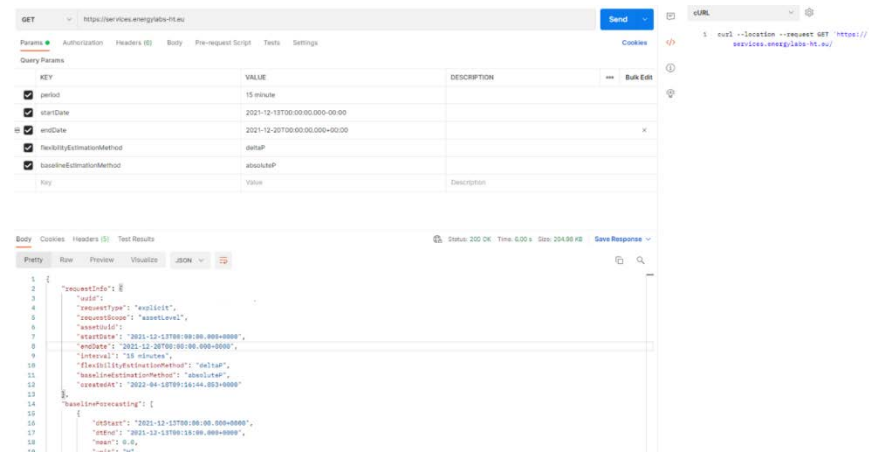


Figure 44. GET request via Postman in the Croatian demo site

On the other hand, the RabbitMQ server oversees handling the interconnections required to exchange event-based data through the use of the AMQP protocol.

RabbitMQ is an open-source message-broker software that originally implemented the Advanced Message Queuing Protocol and has since been extended with a plug-in architecture to support Streaming Text Oriented Messaging Protocol, MQ Telemetry Transport, and other protocols. In this case we are using AMQP protocol to consume messages from queues and then store them into the FUSE's Elastic Search database.

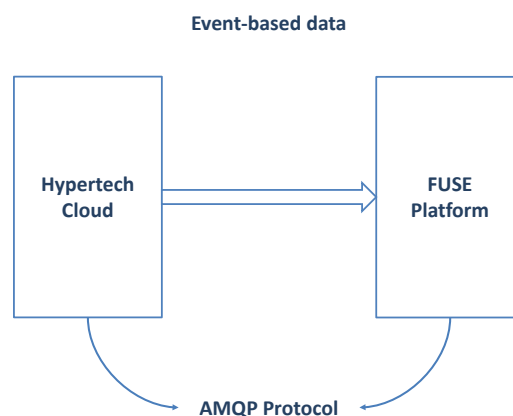


Figure 45. Croatian demo site: AMQP protocol for event-based data

## 5.4. Interoperability in the Italian demonstrator

With the aim to have many measures from the grid in this Italian demonstration site, a STCE-SG provided by SELTA will be installed in a lot of MV secondary substation, both transformation MV/LV and MV costumer substations. In the hydroelectric plants involved in the pilot the STCE will be interfaced with the controller of the plant to regulate the generator.

The whole process of the dispatching platform will be managed by a Smart Grid Controller, located at the SCADA system in ALPERIA. In essence, the algorithms prepared and deployed in the Italian site will need to feed such SCADA and hence present a need for high security features to avoid undesired complications.

Therefore, software modules will be deployed on the pilot to comply with the objectives pursued. Some meaningful data provided from their operative is forwarded to a server integrated with FUSE. There, these data will be employed to perform calculations and via dedicated Graphical User Interfaces (GUIs) provide meaningful data to the end-users.

Hence, to proceed with the interaction partners set up an SFTP server that will be used to exchange such relevant data. It is physically located in the cluster which hosts FUSE platform and reachable in the project's URL [sftp.fuse.flexigrid-h2020.eu](https://sftp.fuse.flexigrid-h2020.eu). Specific credentials must be used to grant users' access to deposit and retrieve data.

Upon performing some tests to confirm the connection works properly via the upload, modification and exchange of sample files via tools such as FileZilla [67], a free and open-source, cross-platform FTP application, consisting of a client and a server, the partners involved in this scenario are in the position to progress in their common interaction. A rough depiction of the interaction proposed in this scenario to foster interoperability appears in Figure 46.

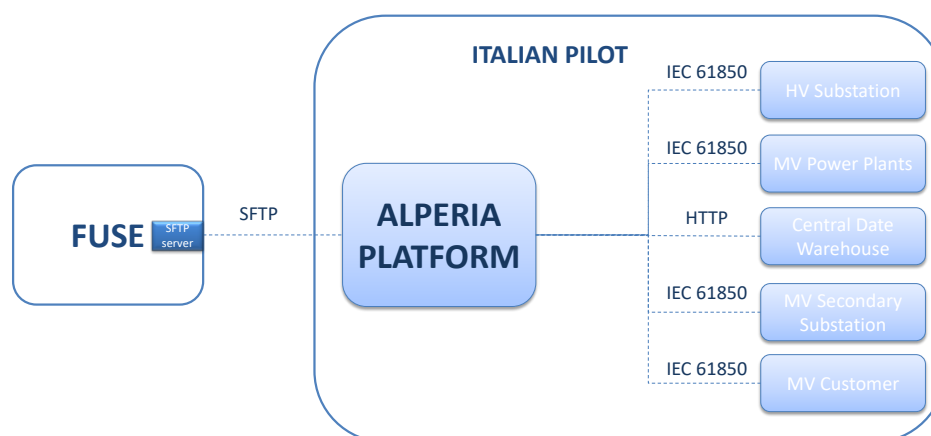


Figure 46. Communications in the Italian demo site

Since the way to proceed with the exchange of relevant data in this pilot is asynchronous, partners agreed on specific times and dates throughout the months of trials execution when these data will be pushed from the Alperia Platform into the SFTP server. Once stored there, FUSE platform will ingest it and carry out the expected calculations that as a result will provide the KPIs required by the pilot owners and users.

## 6. T5.5 FLEXIGRID'S web-based end user interfaces

The current chapter includes a description of the steps taken in the development of the web end-user interface for FLEXIGRID platform, which as expected consist of a common platform for all the users and particular interfaces for each demo-site.

### 6.1. Graphical user interfaces implementation

Depending on the utilization purpose each demonstration site has in mind, the visualization interface provides both historical and real-time information from such site distributed energy (DE) elements (such as generation, consumption, flexibility) along with potential applied demand response (DR) strategies and control set-points per prosumer.

Hence, data shown in the screen will be selective, given users the ability to pick what to present in the screen, also perform meaningful comparisons between series of past data and exploit insights given from data analytics' algorithms running.

The most appropriate graphical representation taking into consideration the overall FLEXIGRID infrastructure will be selected for data presentation. In this context this means that users will be capable to switch between two different tools, where the decision on one or another will rely on the need that every specific pilot has to check dynamic or static Key Performance Indicators (KPIs) that will determine the kind of data they would like to see on the screen. Such tools are namely Kibana [68] and Dash [69]. They both are deeply discussed in the following subsections on this chapter.

As a first step to shape the interface those tools provide, a process ensued to perform a collection of basic requirements and then decide on the general look and feel of the UI.

In addition, authentication mechanisms ensure that access in the developed user interfaces (UIs) are restricted and secure as needed. Upon the original assignment of credentials to pilot owners and relevant participants to gather users' feedback to confirm this solution is valid and satisfies their expectations the need for a refinement appeared: those credentials gave users access to the complete FLEXIGRID infrastructure which is no desirable. Therefore, alternative methods were evaluated, tested and finally implemented.

#### *What is Kibana and how it works in FLEXIGRID*

Kibana is a free and open user interface (UI) that let users visualize data stored in Elasticsearch databases such as the one employed in FLEXIGRID, hosted in FUSE's infrastructure and introduced in D5.6 "FLEXIGRID ICT platform" [6], and navigate the so-called Elastic Stack. Kibana also acts as the UI for monitoring, managing, and securing an Elastic Stack cluster — as well as the centralized hub for built-in solutions developed on the Elastic Stack. Developed in 2013 from within the Elasticsearch community, Kibana has grown to become the window into the Elastic Stack itself, offering a portal for users and companies.

Hence, Kibana provides a wide list of features that will be exploited within the project's scope. For instance, FLEXIGRID users can enjoy aspects related to pure visualizations and data exploration via relevant dashboards and graph analytics. To be precise, Kibana presents information related with those considered as static KPIs in the diverse demonstrator sites.

### What is Dash Plotly and how it works in FLEXIGRID

Dash is the original low-code framework for rapidly building data apps in certain coding languages such as Python. Written on top of Plotly.js and React.js, Dash is ideal for building and deploying data apps with customized user interfaces. It is particularly suited for anyone who works with data and thus it makes sense to employ it in FLEXIGRID context, specifically to depict the results related to the pilot sites' dynamic KPIs.

Dash applications are rendered in the web browser. These apps can be deployed to Virtual Machines (VMs) or clusters in Kubernetes, such as it is the case in FLEXIGRID's infrastructure, and then share them with interested end users through Uniform Resource Locators (URLs).

## 6.2. Data visualization for end users

This section focuses strictly on the provision of diverse examples extracted from the already ongoing pilots. The focus will be on the Greek and Croatian ones, which are the ones exploiting these tools at the time of submission of this report.

To start this recap with Static KPIs and thus the solution in Kibana, which is reachable via <https://kibana.fuse.flexigrid-h2020.eu/>, Figure 47 rescues a sample of Kibana's dashboard appearance when dealing with data related to the Greek pilot. There readers can interact in a customized way with all the data stored in the FUSE Elasticsearch database. They can for example filter by dates or prepare different kind of graphs that capture the relevant data that can be also presented below in pure text format.

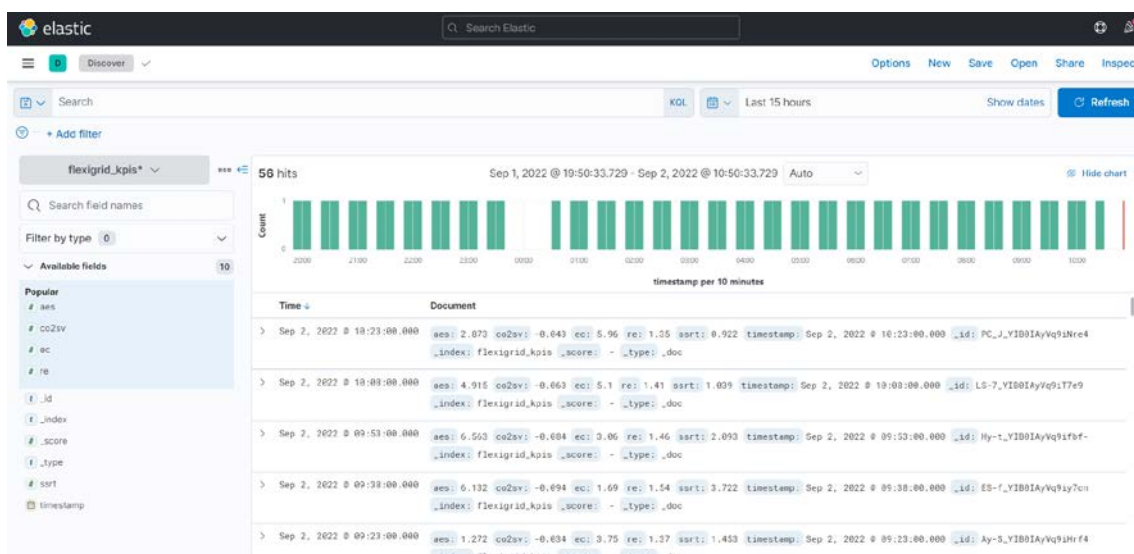


Figure 47. Sample of Kibana Dashboard for Greek Pilot

On the other hand, and referring to Dynamic KPIs, the global link to the dashboard where to check them is <https://dynamic-kpis.fuse.flexigrid-h2020.eu/docs>, and afterwards slight differences appear to accommodate the diverse request related to each demonstrator site.

A couple of examples on this appear in Figure 48 and Figure 49, with some initial calculations and their corresponding graphical representation for the Greek and for the Croatian pilots respectively.

## FUSE FLEXIGRID DASHBOARD

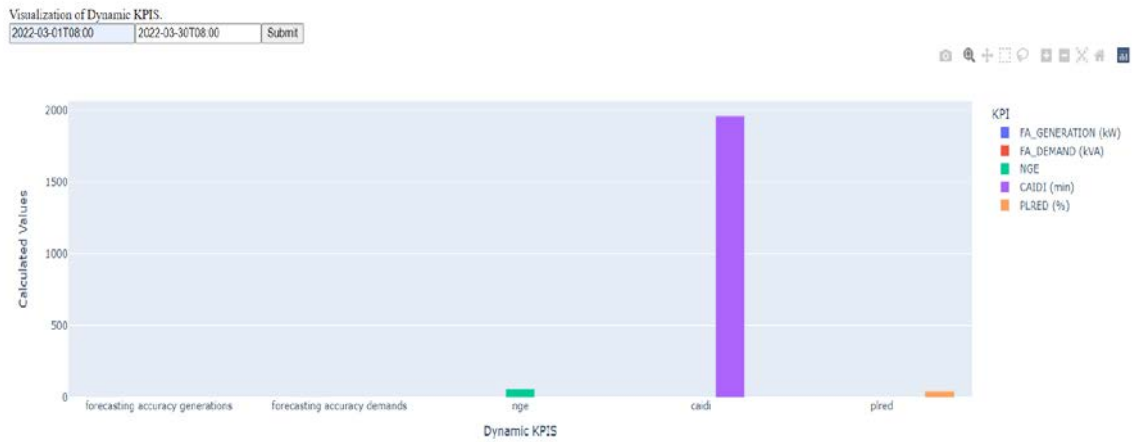


Figure 48. Sample of Dash Dashboard for the Greek Pilot

Regarding the download of the Dynamic KPIs, in this Greek Pilot case the subsequent file would only contain 5 columns, one of them for each KPI subjected to analysis, with one row containing the calculated value of the KPI in the selected interval.

## FUSE FLEXIGRID CROATIA DASHBOARD

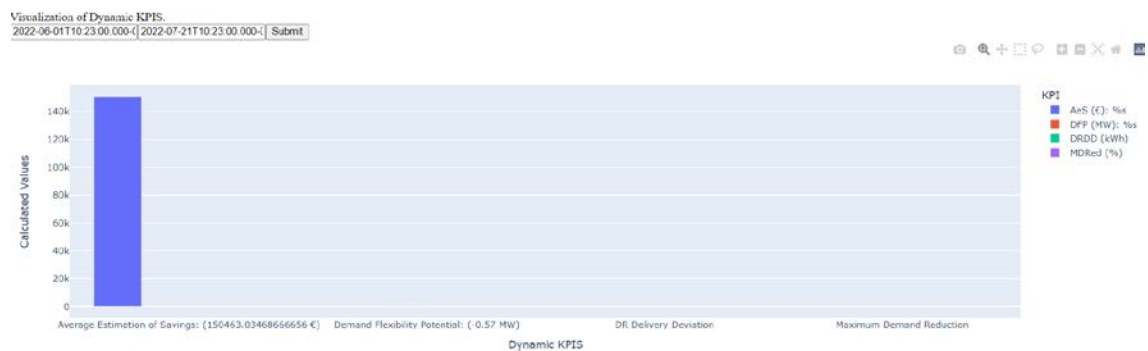


Figure 49. Sample of Dash Dashboard for the Croatian Pilot

Further examples on both GUIs will appear in the WP6 deliverables where results from the piloting phase in each pilot site will be presented.

## 7. CONCLUSIONS

The primary target of this document is to give a definite outline of the work carried out within the scope of WP5, which counts as one of its main objectives the complete definition of FLEXIGRID architecture and all its inherent views, modules and diagrams.

Despite the fact that the work is engraved inside WP5, this report gathers a wide plenty of inputs and viewpoints as, if not, the resultant output would not be legitimate or broadly applicable to the project as a whole. More precisely, this report contains contributions from:

- **Use cases and starting requirements, as characterized in D2.1 report.** This is an essential contribution to the design since FLEXIGRID is an Innovation Action and, along these lines, an undertaking driven by demonstration activities, Use Cases are basic to set the architectural needs and introductory recommendations. In light on top of the Use Cases, this report develops and gives expanded portrayal of layers and modules along with particularization for process diagrams and deployment schemas.
- **Starting portrayal of FLEXIGRID idea and technical approach as introduced in the DoA.** The underlying proposition for architecture as present in the FLEXIGRID proposal has been refined to utilize standard layering and the interfaces and communication among modules have been explained.
- **Specialized vision coming from module developers.** It is not sensible to give an integrated architecture without the vision of every component provider. Thusly, the primary rendition of the logical architecture was given early (see D5.1 [1]) and every one of the important additions and changes were made during the process of composing this report. Thusly, the architecture has been approved by the significant stakeholders, expanding the fittingness of the proposition.

Utilizing all the previously mentioned inputs, this report portrays exhaustively the accompanying views for the FLEXIGRID integrated architecture:

- **Logical view.** Every one of the layers proposed in the design (as a refined form of the one present in the proposition) are given. Individual components are portrayed per layer, zeroing in on inter-dependencies and information flows.
- **Implementation view.** In light of the logical view, all assets and modules given by the FLEXIGRID partners are appointed to the proper layer and portrayed exhaustively. The particularization of the design to be utilized in each pilot site is additionally given.
- **Process view.** This covers the tweaking of Use Case diagrams per pilot, where the most important use cases are featured and particularizations of the outlines are introduced to cover the correspondent use cases on pilot sites.
- **Deployment view.** This view gives the physical layout of parts to be deployed at each pilot site. Along these lines, an initial look into the pilot architecture can be given and coordinated with the one gave as a component of the implementation view.
- **Use case view.** At last, a full arrangement of requirements (both functional and non-functional) are introduced and particularized by use case. This rundown will be utilized in the development phase to validate the usefulness of every module delivered by FLEXIGRID.

All this information can be viewed as the most refreshed architectural version of FLEXIGRID, building on what was presented in D5.1 (M12) [1], including updates and changes as a feature of 1) the regular progression of technical activities and 2) modifications on the pilot sites. Subsequently, presently that pilots and tools are in a more mature stage, the information delivered in D5.2 [2] and presented here reflects precisely the project's approach.

On what refers to T5.2 and following the finalization of the protocols used in FLEXIGRID's demonstration sites, it can be concluded that the current and planned communications available in the project promote interoperability by prioritising the use of recognised standards over proprietary technologies. By referring to FLEXIGRID's ICT architecture description (D5.1 [1] and D5.2 [2]), it can be seen that the only protocols not recognised as a standard in this analysis are PROCOME (Spanish demonstrator) and ENEL Project (Italian demonstrator), but the application in which they are used has not been identified as a potential issue for interoperability or the devices using them are already enabled for alternatively using a standard protocol instead.

In line with this and related to the activities performed in T3.3, the communications between field devices acting as a gateway with capabilities to interact with FUSE (e.g., Energy Box) were analysed. The reason for that is to identify which protocol adaptors are necessary and must be developed to match the communication protocols used for data collection (e.g., Modbus). Atos acted as responsible for the development of these adaptors and for supervising their correct deployment in order to ensure integration

When dealing with the preparation of a CIM for FLEXIGRID, from the point of view of the demonstrators and the components development, the data model aims to establish a common vocabulary to be used by all partners of the FLEXIGRID project and a common knowledge of the distribution power systems.

It is worth to mention that the development using the CIM standard is a dynamic process which is updated every time the respective data need to be updated. To this end, the JSON interfaces which are essential for the data exchange between the components in the FLEXIGRID framework have been outlined and will be fully defined in the final version of this document. This methodology can be also applied to other serialization and communication protocols, for instance, OPC.

The advantage of using CIM standard is not only the complete model of the current electrical grids, but also the ability to model future requirements and to establish relationships between different models. The base of the CIM standard is the semantic techniques as ontologies and ontology alignment that brings powerful tools for modelling and translating.

As the methodology used to add an entity within the FLEXIGRID CIM has been explained, a similar methodology should be used to add those entities which we did not give priority to because they are not important for the project but could be included in the future for scalability.

To guarantee interoperability, those pilots which do not have any simulation software or model the grid in any way, will be allowed to model it with the JSON in case later it is possible to import them into some simulation software to provide the same services that we are lending to the pilots which do use them.

Demonstrator sites that have their own simulation software simply connect them through the Venn diagram intersections as stated above. But to those which do not have software FLEXIGRID

consortium will give priority so that they can model with the project's JSON and import it into any software that service developers use in any demonstrator site, which makes it interoperable.

Going into T5.3, its goal is to describe the cybersecurity mechanisms foreseen by the FLEXIGRID ecosystem. The requirement gathering process, based on the UCs defined and final design of the services offered within the overall pilots considered, lead to the definition of the security principles exposed in the current deliverable. In particular, the risk analysis performed on the overall components involved within FLEXIGRID, by considering the latest threats on smart-grid systems lead to the development of the depicted approach to prevent leakage of sensitive data. In addition, the necessity to avoid the exposure of private company networks and physical systems enforced the approach of maintain autonomous processes for smart-grid devices and remote services through separate and distinct environments.

Considering the possible ways to let pilot's equipment to communicate with the cloud platform and the integrated remote services, even by exploiting the latest solution conceived by the State of the arts with respect to distributed ecosystems, such as the one faced within FLEXIGRID, the possibility to introduce security leaks was a serious issue that required many dedicated meetings and analysis. Since the actual security of a distributed system is strongly affected by the weakest points in the overall people, machines and tools involved, the possibility to underestimate or to not identify promptly a security leakage could lead to disruptive scenarios. Before any attempt to integrate software solutions in such systems is mandatory to go for a software quality assessment, deeply controlled credentials for both users and tools, and highly controlled computers (updates, vulnerabilities, etc.). Since it was not possible to have such control on the overall contributors of the project, the optimal solution agreed by the consortium was first to isolate the sensitive machines and networks directly interacting with smart-grid components. Consequently, each system needs to be independent and able to work in autonomy.

The possibility to exploit remote services through the FUSE platform is an added value that produce value if reachable but is highly important to avoid strict dependency of such systems to the remote ones. This is because, in case of lacks connectivity towards the services built, the pilot's hardware and software need to be able to offer their base services. As a result, the final approach selected to manage in a unified way the various system and services developed within the project lead to the definition of the set of principles described in the current deliverable. In conclusion, the enablers and remote services developed within WP4 are built following zero trust concepts. By following the depicted approach, the pilots considered can work autonomously but, in case of remote services availability, are able to reach optimal solutions with higher degree of confidence.

The selected approach to maintain isolated environments for pilots, cloud and remote services requires redundant controls on the data and its sanitization but is the best solution to prevent threats and malicious behaviour of the sensitive infrastructures involved in FLEXIGRID. In conclusion, the exploitation of the defined mechanism to link pilots and services was agreed by the consortium as the best solution to introduce new features in a safe way. Provided description is firm evidence for the achievement of the defined security goals considering the defined technical objectives.

WP5 also deals with interoperability in FLEXIGRID'S platform. In an ICT context, ISO/IEC 2382-01 [70] defines interoperability as follows: *"The capability to communicate, execute programs, or*

*transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units”.*

Interoperability involves two or more systems that need to be set up to exchange, access, and understand the shared data from the other system. This process requires the syntactic approach, allowing systems to adopt standard data formats and structure protocols. The next step is to use the semantic method, which uses metadata to connect each data element to a controlled and shared set of vocabulary. Once this vocabulary is established, it will be linked to an ontology. An ontology is a data model consisting of concepts and their relationships within a specific domain. By adhering to these standards, components within FLEXIGRID infrastructure can then successfully send relevant information independent of another information system.

The interoperability platform proposed by FLEXIGRID provides tools for assuring their constitutive modules are fully interoperable and can assure the proper data exchanges take place seamlessly. The platform proposes a set of communication protocols, adaptors and data models suitable for the execution of data flows within the platform itself and with the aim to involve upcoming external users from diverse areas in the energy management value chain.

Finally, WP5 must also offer the way for interested users to interact with the platform. Thus, this report captures the process followed in the pursue to get the most suitable web-based interfaces for FLEXIGRID users to interact with the project’s platform, retrieve valuable data and proceed with relevant advanced analysis, not only in a graphical but also in methods purely based on text.

The work around the preparation of graphical user interfaces (GUIs) in FLEXIGRID context must deal with certain constraints that raised consortium partners’ attention once started their development. First and foremost, the difficulties to present values and thus the associated graphical views of the dynamic KPIs considered in each pilot site derived in the need to look for a tool complementary to the one developed in Kibana, which is the ideal one due to its tight relationship with Elasticsearch database, the one deployed in FUSE’s platform. To reach this goal, a solution based on dash Plotly dashboards was implemented and already tested and validated by partners in the pilot sites where trials are currently ongoing.

All in all, the work performed within WP5 covered the aspects related to the technical definition and implementation of FLEXIGRID’s architecture, which will be the basis on top of which the pilots will be able to perform the expected trials that will be properly summarized in the reports emanating from WP6.

## 8. REFERENCES

- [1] FLEXIGRID, «D5.1 FLEXIGRID ICT platform architecture – Month 12,» September 2020.
- [2] FLEXIGRID, “D5.2 FLEXIGRID ICT platform architecture - Month 24,” September 2021.
- [3] FLEXIGRID, “D5.3 Protocols and standards definition,” September 2020.
- [4] FLEXIGRID, “D5.4 FLEXIGRID Common Information Model (CIM),” March 2021.
- [5] FLEXIGRID, “D5.5 Platform cybersecurity mechanisms,” March 2022.
- [6] FLEXIGRID, “D5.6 FLEXIGRID ICT platform,” September 2022.
- [7] FLEXIGRID, “D5.7 Web-based end users' interfaces,” September 2022.
- [8] International Organization for Standardization, «ISO/IEC/IEEE 42010:2011 — Systems and software engineering — Architecture description,» 2011-12-01.
- [9] P. Kruchten, «Architectural Blueprints—The “4+1” View Model of Software Architecture,» *IEEE Software*, vol. 12, nº 6, pp. 42-50, November 1995.
- [10] CEN-CENELEC-ETSI Smart Grid Coordination Group, «Smart Grid Reference Architecture,» November 2012.
- [11] AIOTI, «Definition of a Global Architecture for Smart Grid Applications (COSMAG),» 2019. [En línea]. Available: [https://aioti.eu/wp-content/uploads/2019/03/20181010\\_COSMAG\\_07.pdf](https://aioti.eu/wp-content/uploads/2019/03/20181010_COSMAG_07.pdf). [Último acceso: September 2020].
- [12] Triangle MicroWorks, Inc., «Triangle MicroWorks,» [En línea]. Available: <https://www.trianglemicroworks.com/products/source-code-libraries/iec-60870-scl-pages/overview>. [Último acceso: September 2020].
- [13] IEC – International Electrotechnical Commission, «IEC 60870-5-104 – Part 5-104: Transmission protocols – Network access for IEC 60870-5-101 using standard transport profiles,» June 2006.
- [14] IPCOMM GmbH, «IPCOMM, Protocols,» 2020. [En línea]. Available: [https://www.ipcomm.de/protocols\\_en.html](https://www.ipcomm.de/protocols_en.html). [Último acceso: September 2020].
- [15] ENSOTEST S.L., «Introduction to the IEC 60870-5-104 standard,» ENSOTEST - Energy Software & Testing, 2020. [En línea]. Available: <https://www.ensotest.com/iec-60870-5-104/introduction-to-the-iec-60870-5-104-standard/>. [Último acceso: September 2020].
- [16] ENSOTEST S.L., «INTRODUCTION TO THE IEC 61850 STANDARD,» ENSOTEST - Energy Software & Testing, [En línea]. Available: <https://www.ensotest.com/iec-61850/introduction-to-iec-61850-protocol/>. [Último acceso: September 2020].

- [17] IDE4L project consortium, «Deliverable 3.2: Architecture design and implementation,» 2015.
- [18] E. J. Molina y O. D. Flórez, «Aplicación del Estándar IEC 61850 en los sistemas de protecciones eléctricas para subestaciones de Alta Tensión,» *Revista Clepsidra*, vol. 5, nº 9, pp. 53-59, 2009.
- [19] J. A. R. M. y J. M. O. Moreno, "UP GRID - WP2 - Innovative Distribution Grid Use Cases and Functions," 2016.
- [20] DLMS User Association, «DLMS: Device Language Message Specification,» [En línea]. Available: <https://www.dlms.com/home>. [Último acceso: Septiembre 2020].
- [21] DLMS User Association, Green Book Ed. 9: Excerpt from COSEM. DLMS/COSEM Architecture and Protocols, 2019.
- [22] A. Kovacs, R. Schmidt, D. Marples y R. Morsztyn, «Integrating EVs into the Smart-Grid,» de *13th International Conference on ITS Telecommunications (ITST)*, Tampere, Finland, 2013.
- [23] J. S. Rinaldi, «Real Time Automation,» 13 February 2017. [En línea]. Available: <https://www.rtautomation.com/rtas-blog/why-modbus-has-flourished/#:~:text=Another%20reason%20Modbus%20was%20so,to%20read%20and%20write%20them..> [Último acceso: September 2020].
- [24] Modbus, Modbus Application Protocol Specification V1.vb3, 2012.
- [25] ISO/IEC, Information technology — Message Queuing Telemetry Transport (MQTT) v3.1.1, 2016.
- [26] Endeavor Business Media, LLC, «Electronic Design,» [En línea]. Available: <https://www.electronicdesign.com/technologies/iot/article/21801812/zwave-specifications-go-opensource>. [Último acceso: September 2020].
- [27] L. Oliveira, J. J. P. C. Rodrigues, S. A. Kozlov, R. A. L. Rabêlo y V. H. C. de Albuquerque, «MAC Layer Protocols for Internet of Things: A Survey,» *Future Internet*, 2019.
- [28] A. Singh, «IoT-Point,» [En línea]. Available: <https://iotpoint.wordpress.com/z-wave-tutorial/>. [Último acceso: September 2020].
- [29] Smart Robotic Home, «Smart Robotic Home,» [En línea]. Available: <https://smartrobotichome.com/z-wave-vs-z-wave-plus/>. [Último acceso: September 2020].
- [30] R. Godfrey, D. Ingham y R. Schloming, «OASIS Advanced Message Queuing Protocol,» OASIS Open, 2012.

- [31] R. Cohn, «A Comparison of AMQP and MQTT,» [En línea]. Available: [https://lists.oasis-open.org/archives/amqp/201202/msg00086/StormMQ\\_WhitePaper\\_-\\_A\\_Comparison\\_of\\_AMQP\\_and\\_MQTT.pdf](https://lists.oasis-open.org/archives/amqp/201202/msg00086/StormMQ_WhitePaper_-_A_Comparison_of_AMQP_and_MQTT.pdf). [Último acceso: September 2020].
- [32] J. Postel y J. Reynolds, «RFC 959 - File Transfer Protocol,» October 1985. [En línea]. Available: <https://tools.ietf.org/html/rfc959>. [Último acceso: September 2020].
- [33] M. Belshe, B. R. Peon, I. Google, E. M. Thomson y Mozilla, «Hypertext Transfer Protocol Version 2 (HTTP/2),» May 2015. [En línea]. Available: <https://tools.ietf.org/html/rfc7540>. [Último acceso: September 2020].
- [34] M. Bishop, «Hypertext Transfer Protocol Version 3 (HTTP/3),» September 2020. [En línea]. Available: <https://quicwg.org/base-drafts/draft-ietf-quic-http.html>. [Último acceso: September 2020].
- [35] OPC Foundation, «www.opcfoundation.org,» 2020. [En línea]. Available: <https://opcfoundation.org>. [Último acceso: September 2020].
- [36] EPRI, The Common Information Model for Distribution, Palo Alto, CA: Electric Power Research Institute, 2008.
- [37] IEC – International Electrotechnical Commission, «IEC 61970-301 – Energy management system application program interface (EMS-API) - Part 301: Common Information Model (CIM) Base,» 2020.
- [38] IEC – International Electrotechnical Commission, IEC 61968-11 – Application integration at electric utilities - System interfaces for distribution management - Part 11: Common Information Model (CIM), March 2013.
- [39] IEC – International Electrotechnical Commission, «IEC 62325-301 – Framework for energy market communications - Part 301: Common information model (CIM) extensions for markets,» 12 March 2018.
- [40] IEC - International Electrotechnical Commission, «Framework for energy market communications - Part 351: CIM European market model exchange profile,» 2016.
- [41] FIWARE Foundation, e.V., «FIWARE,» 2020. [En línea]. Available: <https://www.fiware.org>. [Último acceso: September 2020].
- [42] J. M. Cantera, F. Galán y T. Jacobs, «FIWARE-NGSI v2 Specification,» [En línea]. Available: <https://fiware.github.io/specifications/ngsiv2/stable/>. [Último acceso: September 2020].
- [43] ETSI, «ETSI GS CIM 009 V1.3.1 - Context Information Management (CIM); NGSI-LD API,» 2020.
- [44] FIWARE Foundation, e.V., «Fiware data models,» 2020. [En línea]. Available: <https://fiware-datamodels.readthedocs.io/>. [Último acceso: September 2020].

- [45] M. Balijepalli and K. Pradhan, Review of Demand Response under Smart Grid Paradigm, 2011.
- [46] OpenADR Alliance, «OpenADR Alliance,» [En línea]. Available: <https://www.openadr.org/>. [Último acceso: September 2020].
- [47] IEC – International Electrotechnical Commission, «IEC 62746-10-1 – Systems interface between customer energy management system and the power management system - Part 10-1: Open automated demand response,» *International Electrotechnical Commission*, 2018.
- [48] ETSI, «ETSI TS 103 264 - SmartM2M; Smart Appliances; Reference Ontology and oneM2M Mapping,» 2017.
- [49] K. Kimani, V. Oduol y K. Langat, «Cyber Security Challenges for IoT-based Smart Grid Networks,» *International Journal of Critical Infrastructure Protection*, vol. 25, 2019.
- [50] E. M. Zakaria, K. Naima, E. G. Hassan y E. G. Hamid, «Cyber-security in smart grid: Survey and challenges,» *Computers & Electrical Engineering*, vol. 67, pp. 469-482, 2018.
- [51] Z. Muhammed y R. D. Gunduz, «Cyber-security on smart grid: Threats and potential solutions,» *Computer Networks*, vol. 169, p. 10794, 2020.
- [52] «CyberSEAS,» [En línea]. Available: <https://cyberseas.eu/>.
- [53] «ELECTRON,» [En línea]. Available: <https://electron-project.eu/>.
- [54] «EnergyShield,» [En línea]. Available: <https://energy-shield.eu/>.
- [55] «PHOENIX,» [En línea]. Available: <https://phoenix-h2020.eu/>.
- [56] «SDN-microSENSE,» [En línea]. Available: <https://www.sdnmicrosense.eu/>.
- [57] «BRIDGE,» [En línea]. Available: <https://www.h2020-bridge.eu/>.
- [58] Visual Capitalist, “The Most Significant Cyber Attacks from 2006-2020, by Country,” Specops Software, 10 May 2021. [Online]. Available: <https://www.visualcapitalist.com/cyber-attacks-worldwide-2006-2020/>.
- [59] «KISS,» [En línea]. Available: [https://en.wikipedia.org/wiki/KISS\\_principle](https://en.wikipedia.org/wiki/KISS_principle).
- [60] «traefik,» traefiklabs, [En línea]. Available: <https://traefik.io/>.
- [61] «Rancher,» [En línea]. Available: <https://rancher.com/>.
- [62] FLEXIGRID, “D6.2 Spanish demo-site UCs and start-up report,” December 2021.
- [63] FLEXIGRID, “D6.3 Greek demo-site UCs and start-up report,” December 2021.
- [64] FLEXIGRID, “D6.4 Croatian demo-site UCs and start-up report,” December 2021.

- [65] FLEXIGRID, "D6.5 Italian demo-site UCs and start-up report," December 2021.
- [66] "Postman API platform," 2022. [Online]. Available: <https://www.postman.com/>.
- [67] «FileZilla - The free FTP solution,» 2022. [En línea]. Available: <https://filezilla-project.org/>.
- [68] "Kibana: Explore, Visualize, Discover Data," Elasticsearch B.V., 2022. [Online]. Available: <https://www.elastic.co/es/kibana/>.
- [69] "Dash Documentation & User Guide," Plotly, 2022. [Online]. Available: <https://dash.plotly.com/>.
- [70] International Organization for Standardization, "ISO/IEC 2382-1:1993 Information technology Vocabulary - Part 1: Fundamental terms," [Online]. Available: <https://www.iso.org/standard/7229.html>.
- [71] FLEXIGRID, "D2.1 Demo-Sites description and boundary conditions report," June 2020.
- [72] S. N. Islam, Z. Baig y S. Zeadally, «Physical Layer Security for the Smart Grid: Vulnerabilities, Threats, and Countermeasures,» *IEEE Transactions on Industrial Informatics*, vol. 15, nº 12, pp. 6522-6530, 2019.
- [73] K. M. Rossella Mattioli, «Communication network interdependencies in smart grids - Annexes,» ENISA, 2015.
- [74] «OpenAPI,» [En línea]. Available: <https://swagger.io/specification/>.
- [75] FLEXIGRID, "D6.1 Demonstration and monitoring plan," October 2021.

## 9. ANNEX 1 – FLEXIGRID logical architecture per demonstrator

The purpose of this annex is to describe how each demonstrator is applying FLEXIGRID's reference logical architecture according to their requirements. More specifically, these subsections target the peculiarities found in the information, function and business layers from a functional perspective.

### 9.1. Spanish Demonstrator

The Spanish Demonstrator is divided in 3 scenarios: Scenario 1, aimed at demonstrating upgraded substations and testing grid automation and control algorithms; and scenarios 2 and 3, focused on demonstrating grid protection, fault location and self-healing algorithms.

#### *Scenario 1*

From an information layer perspective, the data are going to be collected in this scenario in batches, using FUSE's ETL module. The reason behind this is the limited access to VIESGO's platform due to their security policy.

A diagram of the software modules related to this scenario can be found in Figure 50. These modules are located in the upper layers of the reference architecture, inside the region labelled as *S6 – Forecasting and grid operation*. The idea is to train two forecasting models for generation data with different frequencies (1 minute and 24 hours), and two equivalent models for consumption data. Afterwards, by using the output of the resulting models (inference based on latest batch data), two different modules will calculate the optimal set points for flexible assets. LINKS's predictive optimisation algorithm will calculate the optimal setpoints for efficient operation, while CIRCE's flexibility assets operation will calculate the setpoints in case of contingencies to maintain the integrity of the grid. Consequently, a grid condition discriminator will evaluate the coming status of the grid using the output of the 1 min forecast module and decide which setpoints to use, giving priority to CIRCE's module in case of contingency. For more information about these modules or the nature of the flexible assets they control, refer to D4.3.

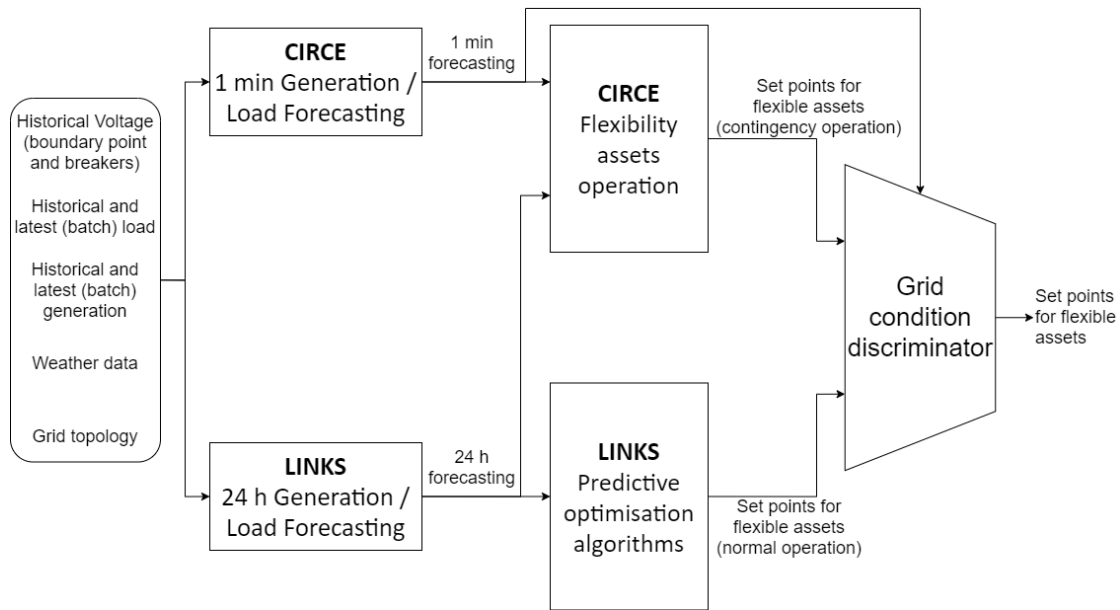


Figure 50. Diagram of software modules used in Spanish demonstrator's scenario 1.

### Scenarios 2 and 3

Scenarios 2 and 3 will be using the same software modules. This is because scenario 2 will be used for testing the algorithms with artificially generated faults, while scenario 3 is meant to validate the results in real operation. The data for these scenarios will also be provided in batches due to VIESGO's security policy.

The software modules involved in this scenario are shown in Figure 51. In the reference architecture, they are located in the region labelled as *S5 – Fault location and self-healing*. As seen in the diagram, both algorithms take the same inputs: active and reactive power measured at the boundary point, the status of the breakers before the fault occurred, the currents and voltages of the Intelligent Electronic Devices (IED) deployed in the distribution grid, LV power consumption (from a concentrator at the MV substation) and the grid topology. The output of the fault location algorithm will be the location of the fault in terms of the grid segment and the distance to the fault. The output of the self-healing algorithm will be the resulting status of the breakers (open/close) to recover from the fault.

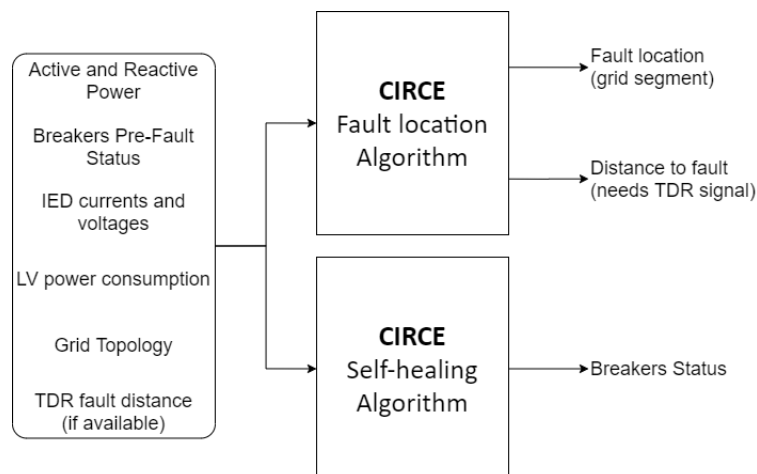


Figure 51. Diagram of software modules used in Spanish demonstrator's scenarios 2 and 3.

## 9.2. Greek Demonstrator

From an information layer perspective, the Greek demonstrator will provide data to FUSE platform in real time.

Regarding the upper layers of the architecture, the modules involved can be observed in Figure 52. In this case, two different models for demand and generation forecasting will be trained and infer the upcoming demand/generation making use of current data, weather information and a local special calendar. The forecast will follow a rolling horizon with 24-hours length and granularity of half an hour. Updates will occur based on the scheduling table and/or error between observed and forecasted values. The resulting forecasts, some technical information from the flexible assets available and pricing inputs will then be used to calculate the optimal schedule for the flexible assets for peak shaving and reduction of total cost of the energy. The flexible assets schedule will be used to control the output from the batteries and cut-off signals for EV charging stations.

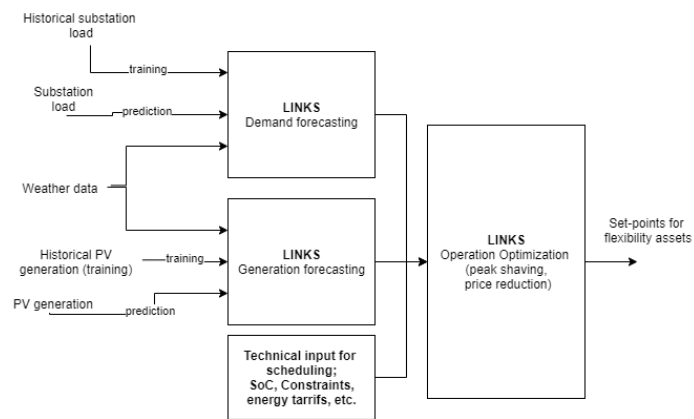


Figure 52. Diagram of software modules used in the Greek demonstrator.

On the other side, the pricing strategies analysis module will take the same inputs as the congestion management and peak shaving module and will calculate the benefits of the ancillary service. All these modules can be found in FLEXIGRID's reference logical architecture in the region labelled as S7 – *Grid congestion management*.

## 9.3. Croatian Demonstrator

Croatian demonstrator will be conducted in an apartment building (LV customers). For the LV customers, data will be collected to calculate control strategies for their flexibility resources (i.e., available controllable thermal loads, such as HVAC systems and hot water boilers) so the building acts as a Virtual Energy Storage (VES). LV customers are involved in use case 6. For use case 5 the urban district in city of Zagreb was chosen.

Due to privacy issues concerning this demonstrator, the data collected from participating customers will not be stored in FUSE. Instead, FUSE will be used to only harmonise the data before it reaches its destination.

In Figure 53, one can see the software modules corresponding to the distribution network flexibility and protection schemes coordination. They all take as common inputs static distribution network technical data, and the historical consumption profiles at the beginning of the feeders and the local measurement devices. Apart from that, each of them takes an

additional input to fulfil their purpose. The colours of the arrows have no special meaning, except for easier tracking.

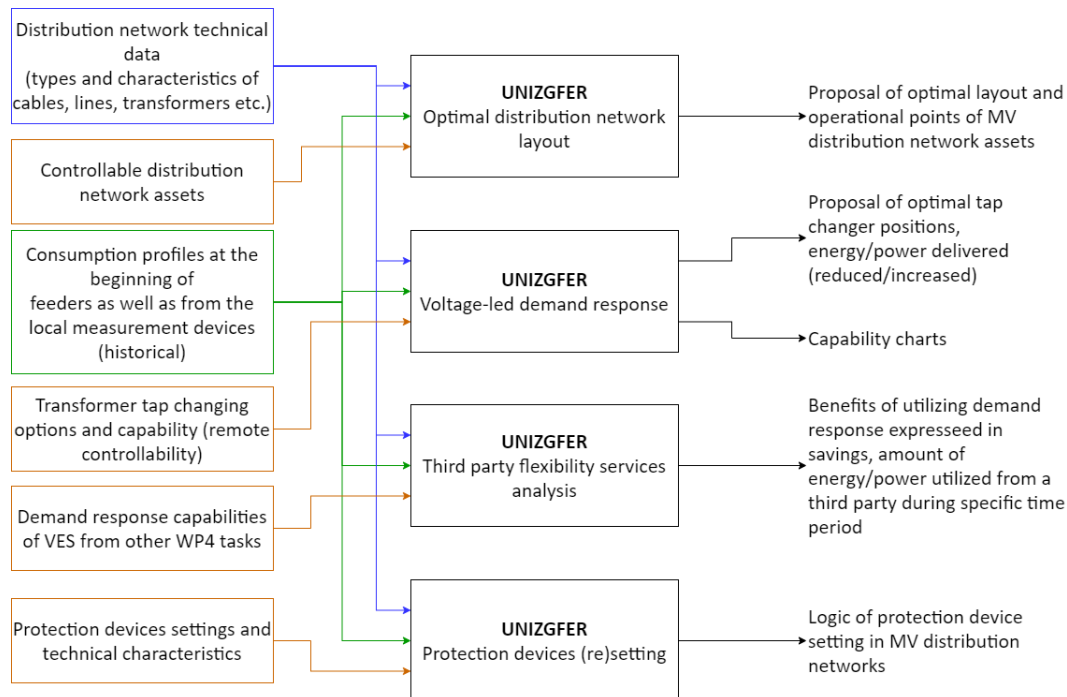


Figure 53. Diagram of software modules used in the Croatian demonstrator's distribution network flexibility and protection schemes coordination.

The optimal distribution network layout defines the optimal MV network layout to comply with technical requirements and ensure maximum reliability of supply, while minimizing network losses. It takes, as an additional input, the controllable distribution network assets available at the substations. This module is considered as FLEXIGRID's solution S7 (Grid congestion management) in the reference logical architecture in Figure 3.

The voltage-led demand response module takes as an additional input the transformers tap changing options and capabilities to exploit the active power-voltage correlation, and defines the new operational point requested by the TSO. It needs to ensure that technical constraints are met and that the service is met without increasing the DSO's losses. In order to ensure the aforementioned technical constraints, this module will create capability charts for the DSO to use as a tool to signal the TSO about those constraints. This module is considered part of FLEXIGRID's solution S7 in the reference logical architecture.

The third-party flexibility services analysis module will analyse the benefits of using demand response (VES or any other commercial flexibility provider such as battery storage) and provide an output of the savings and amount of energy or power used from a third party. In the reference logical architecture, this module can be found as part of FLEXIGRID's solution S7.

The protection devices (re)setting, as its name implies, takes the technical characteristics of such devices as an input and rechecks their settings after operational decisions (dispatch) are made in the previous three modules. If these settings need corrections, they are performed, and the relays are reset to them.

On the other hand, Figure 54 shows the software modules corresponding to the thermal energy storage optimisation (all located in the reference logical architecture within the S8 region). As observed in the diagram, it follows a sequential flow.

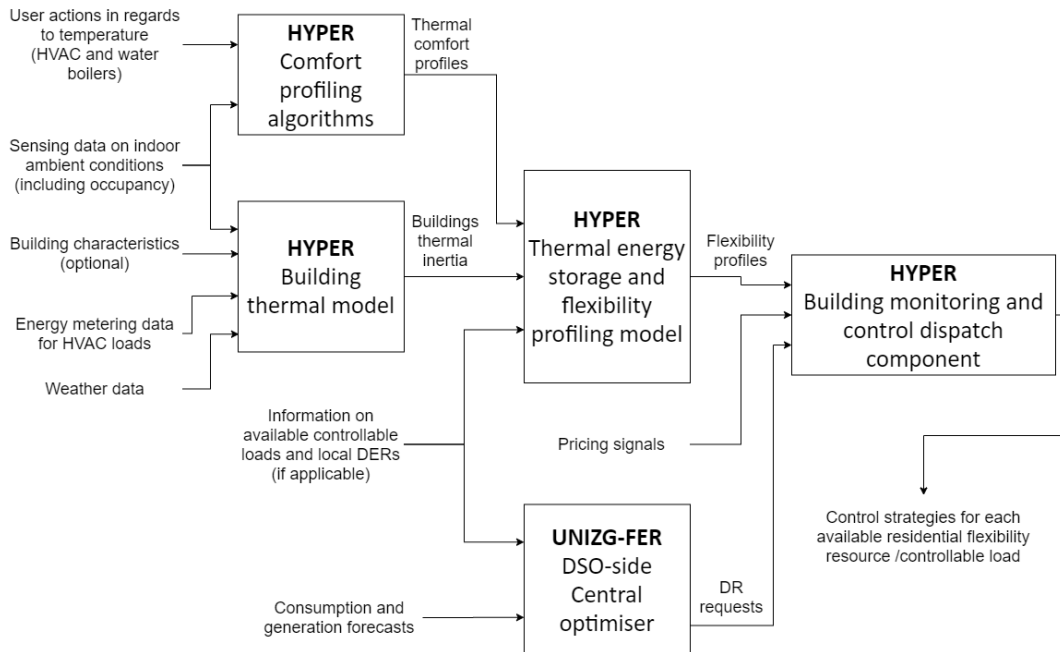


Figure 54. Diagram of software modules used in the Croatian demonstrator's thermal energy storage optimisation.

First to intervene are the comfort profiling algorithms and the building thermal model. The former takes the users' actions over HVAC systems and water boilers (i.e., thermal loads) and data collected by multi-sensors regarding ambient conditions (including occupancy). Afterwards, by using the aforementioned data and pieces of information, it calculates the thermal comfort boundaries that are representative of the occupants of a specific building zone that includes controllable thermal loads. The latter considers, apart from the ambient conditions, the building characteristics, energy metering data from available controllable thermal loads, and weather data. Subsequently, it creates physics-based, data-driven thermal inertia models to simulate the thermal behaviour of buildings. This methodology is suitable for building areas and hot water storage tanks. The extraction of flexibility derives from parametric models of the thermal storage properties of thermal storage equipment (e.g.: capacity, retention period) and from their electrical response characteristics (e.g.: response time, ramp up and ramp down times, rated and actual power or energy consumption).

The output of the previous two modules and additional available information about controllable loads—capacity of DHW tanks, HVAC temperature ranges, etc.—will then be used as an input to the thermal energy storage and flexibility profiling model. This model will define the pre-heating/pre-cooling flexibility that individual buildings can offer in the context of optimised demand response strategies.

Parallel to the previous module, a predictive optimisation algorithm will create demand response requests based on the information about the controllable loads and the forecasts for consumption and generation at grid level.

Lastly, the building monitoring and control dispatch component will break down the DR requests of the previous module to control strategies of the available devices (e.g.: HVAC, DWH). It will

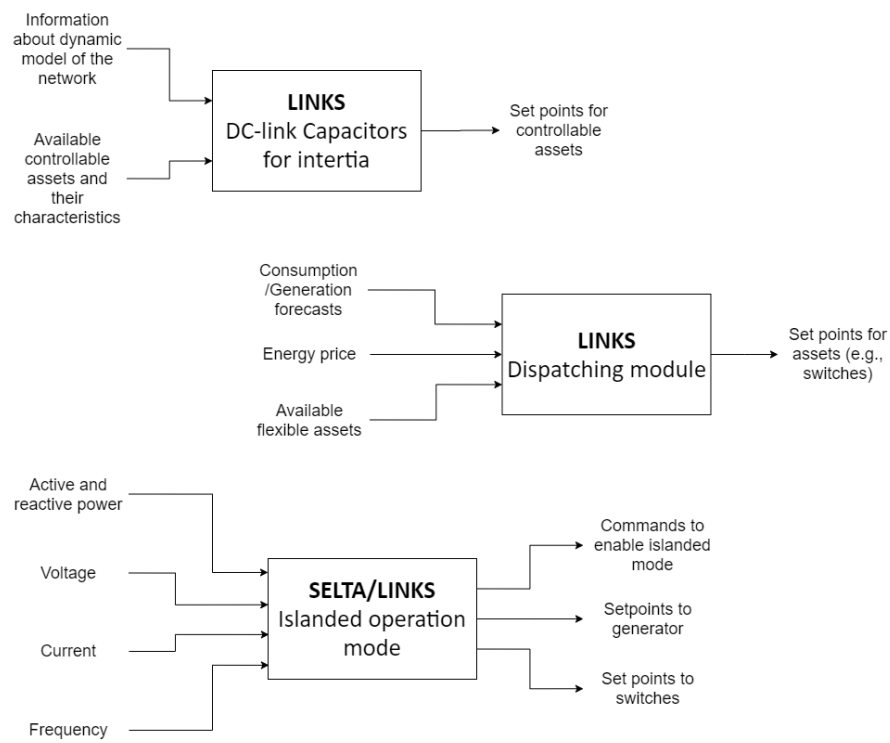
also take into consideration the flexibility profile from the profiling model and pricing signals coming from the DSO platform.

#### 9.4. Italian Demonstrator

From an information layer's perspective, this demonstrator will not provide real-time data to FUSE. The SCADA system involved within the Italian pilot is a crucial element of the supply grid. This cyber security aspect regulated by the European directive NIS (Network and Information Security) and its Italian transposition includes the DSOs in the critical infrastructures. Consequently, the risks of opening network interfaces of a single machine, could lead to serious issues. The set of existing malicious attacks could exploit known and unknown vulnerabilities, even unrelated to the software modules developed and secure within the project. Because of the sensitivity of such deployment, the consortium agreed on performing the trials exploiting software modules deployed within the pilot. During the testing phase, the data will be collected and then delivered via SFTP to the FUSE infrastructure to ease KPI calculation through a dedicated GUI.

Figure 55 shows the diagrams of the three independent modules available at this demo. The first one is an algorithm that calculates virtual inertia for controlling assets to keep frequency in a secure range. This will be mostly simulation rather than an integrated control method. The developed concept will then be used for T4.6 (Islanded mode operation, which results in the last module of Figure 55). It is located in FLEXIGRID's solution S5 area in the reference logical architecture for its relationship with grid stability and protections.

The remaining two modules, located in the solution S6 area of the reference logical architecture, have dispatching functionalities: the one simply called *dispatching module* will monitor and control the available assets to handle congestion and power quality issues. This module is related to the *steady-state* behaviour of the system. Likewise, the *islanded operation mode* module will secure operation in islanded mode from the transmission system. It is worth mentioning that the islanded operation mode module will not deploy control logic in the cloud. It will operate locally and send data to FUSE off-line for the KPIs calculation.



*Figure 55. Diagram of software modules used in the Italian demonstrator.*

## 10. ANNEX 2 – FLEXIGRID deployment view per demonstrator

### 10.1. Spanish Demonstrator

This subsection describes the deployment details of the two sets of scenarios specified for this particular demonstration site.

#### *Scenario 1*

As shown in Figure 56, scenario 1 comprises the deployment of a new substation of the future (targeted in T3.1) and the retrofitting of an already available substation to include connectivity and control capabilities using CIRCE's Energy Box (targeted in T3.3), which will be communicating with FUSE using RESTful Web Services. Additionally, each substation will have a nearby weather station associated to it that will provide meteorological data to VIESGO's platform using Modbus TCP.

Furthermore, the retrofitted substation will be supplying electricity to the LV customers in which the new generation smart meters targeted in task T3.2 are going to be deployed. These smart meters will be integrated using DLMS over Power-Line Communication (PLC). Subsequently, the data will be collected using a concentrator installed in the retrofitted substation and then sent to a Head-End System in VIESGO's platform via FUSE, reachable through an FTP server.

Due to the substation of the future participating in this scenario being one of the main objectives for FLEXIGRID (solution S1), the specific deployment view of such substation is shown in Figure 57. Notice that there are three sections dedicated to medium voltage automation (*Aut. MT*), a smart low-voltage distribution panel (*ADDIBO*), and a smart transformer (*SMART TRAF0*). Also notice that the elements not described here are user interfaces (labelled as *Web Access*) and data platforms: FUSE and specific components of VIESGO's platform—a SCADA system, an asset management platform called ARM, and a telemetry system called STG. The channel for the communication between the substation and these platforms is GPRS, as shown in Figure 56.

Lastly, the data collected will be shared with the consortium in batches, from VIESGO's platform to FUSE. The communication protocol chosen for that purpose will be FTP.

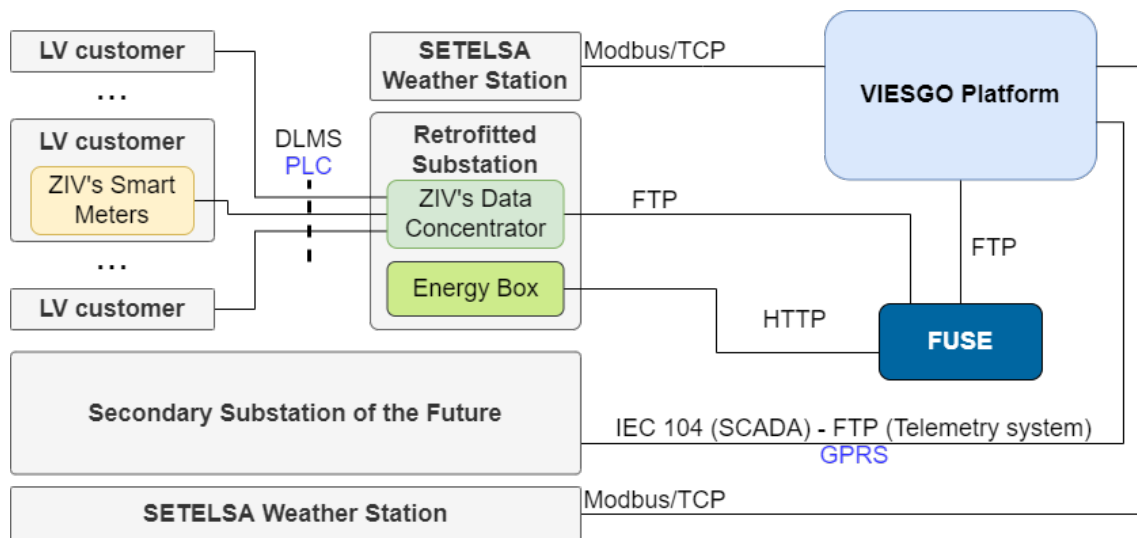


Figure 56. Deployment view of scenario 1 in the Spanish demonstrator.

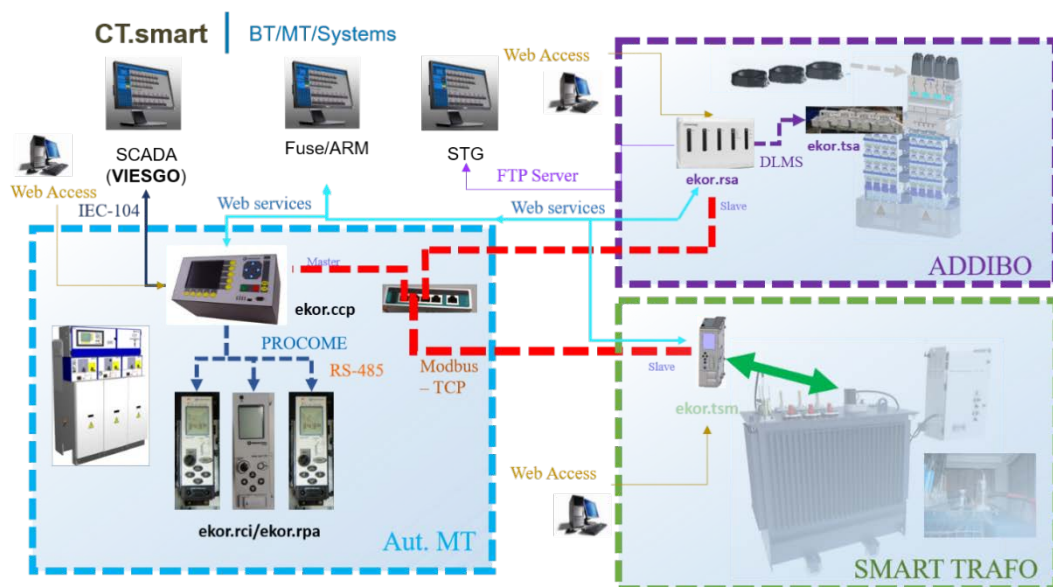


Figure 57. Deployment view of ORMAZABAL's substation of the future.

### Scenarios 2 and 3

From an ICT perspective, scenarios 2 and 3 have an identical deployment view despite being different stages in reality. Scenario 2 corresponds to the substation of Toranzo, in which artificial faults will be generated; and scenario 3 corresponds to the substation of Meira, in which the detection and self-healing algorithms will be tested against real faults. For more details about these scenarios, refer to section 2.1 of deliverable D2.1 (*"Demo-sites description and boundary conditions report"*) [11].

The deployment view for these scenarios is shown in Figure 58 depicts a deployment view based on the current planning of WP4 and WP3 tasks, defining communication protocols and channels.

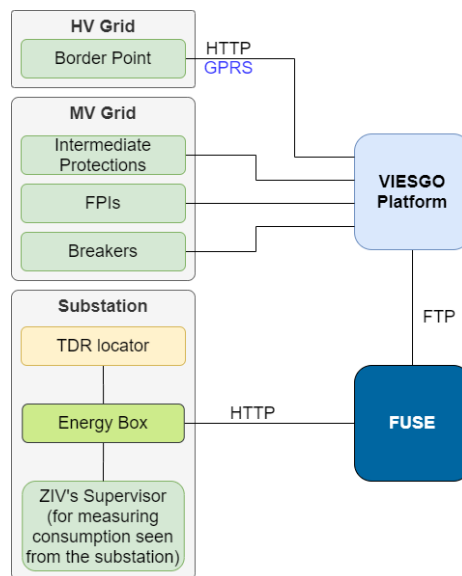


Figure 58. Deployment view of scenarios 2 and 3 in the Spanish demonstrator.

## 10.2. Greek Demonstrator

The deployment view for this demonstrator can be seen in Figure 59. The three relevant bungalows will have the same setup: A battery, a PV installation and a smart meter connected to a hybrid inverter, and another PV installation connected to a non-hybrid inverter. The connection of the battery and the smart meter to the hybrid inverter utilizes Modbus RTU protocol using an RS485 interface. Data pertinent to the action of batteries, PV installations and smart meters are aggregated by the respective inverters and are in turn retrieved from a FLEXIGRID S6 Energy Box via Modbus TCP using the Local Area Network (LAN) of the demo site. The same pathway is utilized to push battery control signals produced by the respective application and channelled through Fuse platform and the Energy Box back to the inverters. Data collection from the analysers installed in the substation as well as control signals to the EV charger relay will be channelled to the same Energy Box through a Modbus TCP – Ethernet / RS485 interface, which establishes a local Modbus RTU network among those devices.

Lastly, the data will be sent from the Energy Box to FUSE using HTTP protocol. Once in FUSE, data will be harmonised and redirected to the corresponding software modules contained on it, as well as to the UIs developed in Task 5.5.

It is important to highlight that for this demo site, the channels are bidirectional, meaning that commands will also be sent from the cloud platform down to the controllable assets (batteries and EV charging station) to change their behaviours based on the outputs of the congestion management and peak shaving algorithms.

The weather data collected for this demo site will be taken from OpenWeatherMap, an external weather API using RESTful web services.

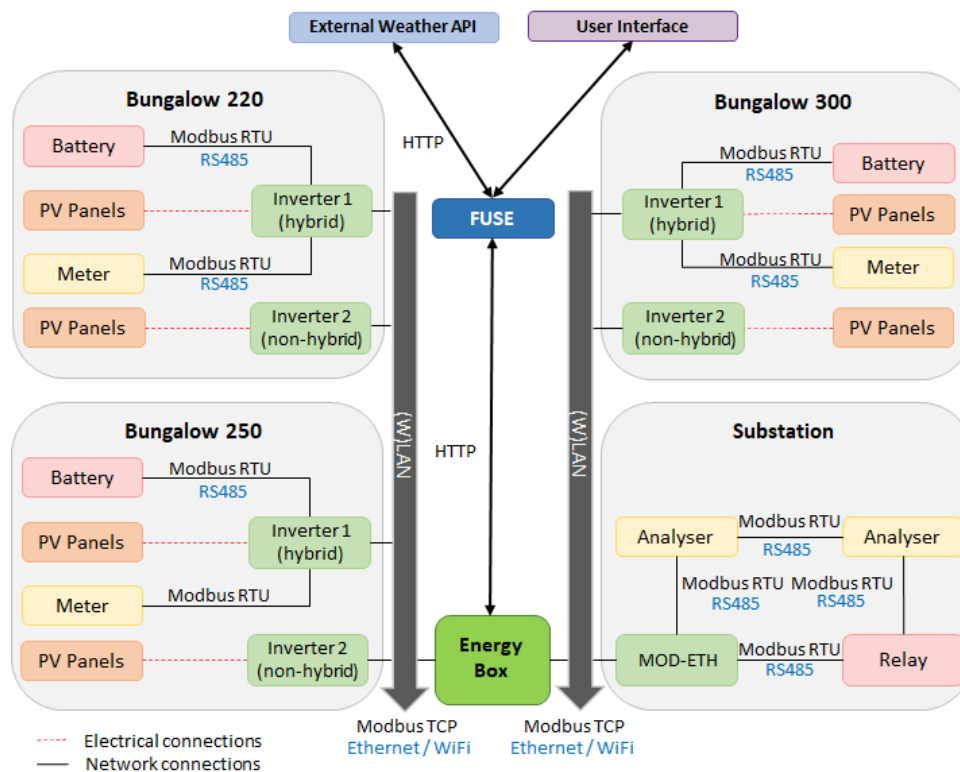


Figure 59. Deployment view of the Greek demonstrator.

### 10.3. Croatian Demonstrator

The Croatian demonstrator takes place in various locations: four MV substations distributed in three different distribution networks, and an apartment building.

Figure 60 shows a block diagram of the substations taking part in this demonstrator. The substations tagged as ‘Demo site’ are the main MV substations in which the operations of the demonstrator are under development. All other substations—labelled only with their substation ID—correspond to the HV substations connected to the same distribution network. These HV substations are represented in the diagrams because some data are also being collected in them.

Regarding the communications in all substations with intelligent electronic devices, they correspond with the communication scheme shown for the DSO SCADA system described in section 4.9 in deliverable D2.1 [11], using one of several options for communication channels (GPRS, RipEX or WiMAX). Besides, communication from the electricity meters to HEP-ODS’s platform is made using DLMS over optical fibre.

At the end of the data flow, the communication between HEP-ODS’s platform and FUSE will be made using RESTful web services. Once there, the data will be harmonised and forwarded to the software modules without being stored in any database whatsoever. This will also be communication channel used for sending to HEP-ODS platform the (harmonised) data collected from customer’s premises by Hypertech, as shown in Figure 61.

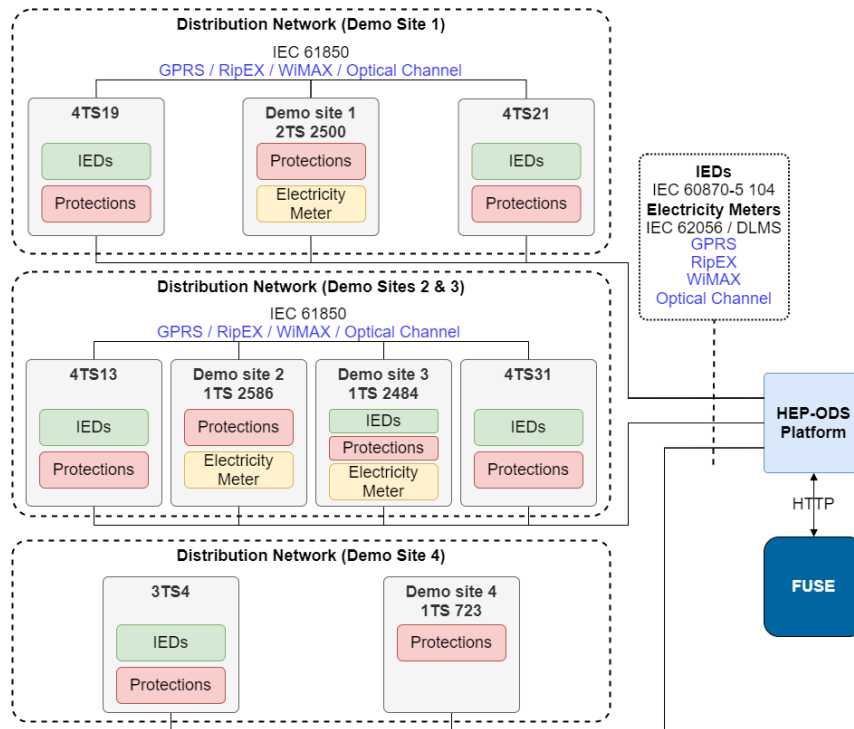


Figure 60. Deployment view of the substations taking part in the Croatian demonstrator.

Regarding the electrical connections to the distribution network, demo sites 1, 2 and 3 were in the end discarded due to the fact they were not suitable for the installation of VTES sensors; while on the other hand, demo site 4 supplies electricity to LV customers—more precisely, to an apartment building. This apartment building is also subjected to operations within the Croatian demonstrator, since it is where the software modules are going to be tested for VES. Therefore, the data provided by the HEP-ODS's platform in this diagram is related to the substation labelled as 'Demo site 4'. Both the data coming from the DSO's platform will be provided to FUSE via RESTful web services.

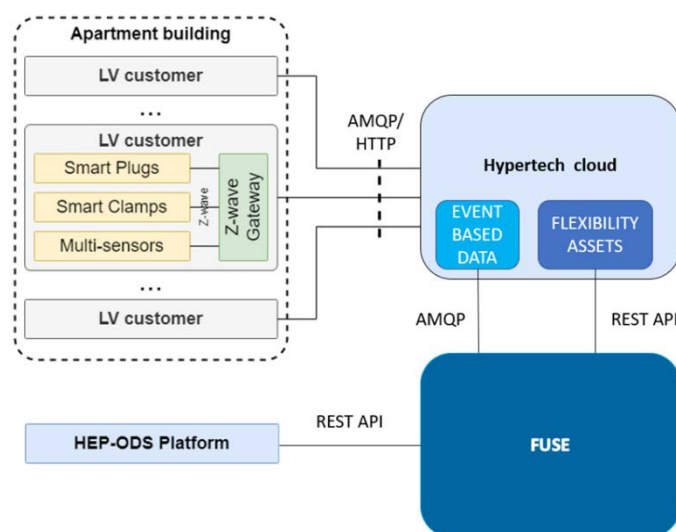


Figure 61. Deployment view of the substations taking part in the Croatian demonstrator.

Inside the apartment building, LV customers' data will be collected using smart metering and sensing devices that communicate using Z-wave to a gateway that will communicate the data to

Hypertech's cloud using HTTP and AMQP. Afterwards, the data will be processed in order to calculate the flexibility profiles, which will be sent to FUSE using HTTP. The remaining event-based data (e.g., metering, controlling and sensing) necessary to execute the predictive optimisation algorithm will be obtained by FUSE using AMQP protocols.

#### 10.4. Italian Demonstrator

The deployment view of the Italian demonstrator is depicted in Figure 62. The main site of operations is CP-UW SARENTINO NEW, as the HV substation for the distribution grid. From there, a total of 6 MV lines distribute electricity to the region. All the data necessary to test the software modules will be collected in ALPERIA's platform, where the whole data from the sites involved in the pilot will be sent, stored and processed. The data for the KPIs calculation will be transmitted to FUSE off-line using an SFTP server.

Separately, the data generated from the second-generation smart meters in LV customers will be sent to a data concentrator at their respective MV or LV substation using a proprietary ENEL protocol—ENEL Project based on SITRED-ST protocol (Kaifa CE)—over PLC. In turn, the data is sent to a central warehouse where ALPERIA's platform will get it using Web Services.

Meanwhile, the information being generated at substations, power plants, and MV customers will be sent to ALPERIA's platform via IEC 61850 over GPRS using SELTA's RTU. The same protocol is used for communication between MV power plants in islanded operation mode, using SELTA's RTU as well.

Regarding MV plants, it is also important to mention that these generators, which correspond to the set of hydroelectrical power plants distributed along the valley mentioned in D2.1, are bound to installing a device to provide the measurements of the plant to the TSO (the device is labelled in the diagram as CCI).

On the other hand, except for the Sarentino HV substation which uses IEC 61850, internal communications within other premises are diverse, but Modbus is the dominant protocol. At this point, it is worth mentioning that the meters shown in MV substations and customers are generic measuring devices used for monitoring these premises, but for the scope of this project only electrical measures are used.

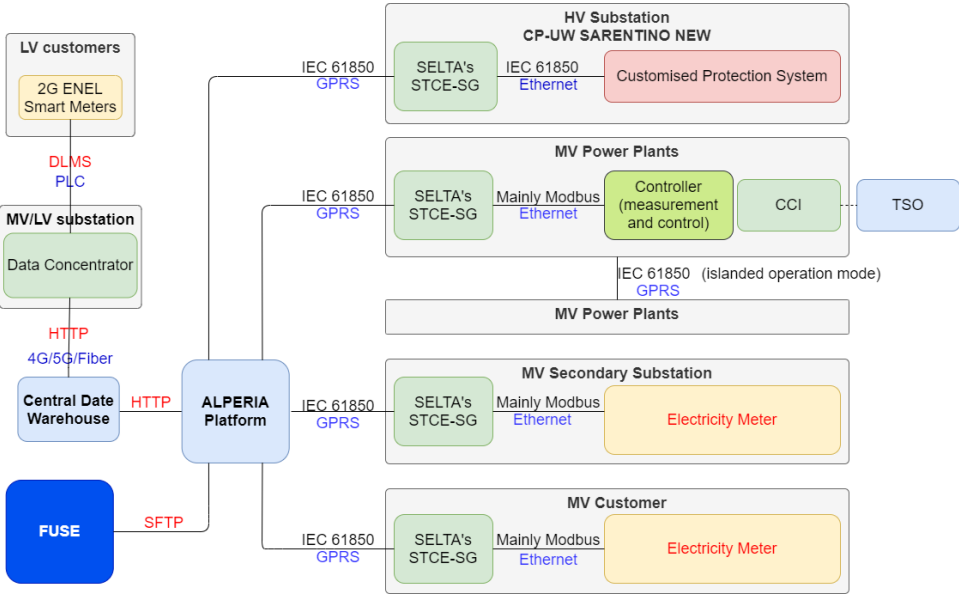


Figure 62. Deployment view of the Italian demonstrator.

## 11. ANNEX 3 – FLEXIGRID process view

### 11.1. Demonstrators and Sequence Diagrams

In this chapter, demonstrators will be summed up and their step-by-step processes, as depicted in D6.1, will be called attention to and supplemented with the sequence diagrams. Subsequently, this part will utilize the information previously introduced in the aforementioned deliverable to give the first iteration on sequence diagrams, which are a graphical method to reflect how each use case can be feasible and tested through the interaction of FLEXIGRID modules, actors and general items in the architecture, as introduced in previous sections.

For every demonstrator, its name, general portrayal, highlighting the quantity of scenarios recognized and the newly produced sequence diagrams are introduced.

### Spanish Demonstrator

This demonstrator will include use cases 1 and 2 and their related trials. Table 4 introduces them all and provides a glimpse on relevant topics such as what will trigger the event that causes that trial and the associated pre and post conditions.

Table 4: Use Cases and trials of Spanish Demonstrator

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC01T01	UC1	Smart transformer self on load commutation	OPA	<p>1.- Scada gives the order (remote control) to trigger (up/down) the on load tap changer</p> <p>2. - Autonomous operation based on voltage measurements to trigger (up/down) the on load tap changer.</p> <p>Supervision of the voltage values and on load tap-changer performance (remote or local)</p>	Network operating normally and the voltages of installation of the final clients are between normalized limits (under /over).	A tap change is made to return the voltage of the installations to normalized values
UC01T02	UC1	ADDIBO performance	OPA	Periodic execution once launched.	Network operating normally, devices connected with MV RTU,	Obtaining the updated information of the LVB of the Secondary Substation

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
				On-demand	LVB RTU active and connected with devices installed in the LVB and communications with central server active.	devices: instantaneous measurement values, log of historical and events, alarms.
UC01T03	UC1	Forecasting algorithm	LINKS	Automatic service run, based on the user setting (e.g. every 15 minutes recalculates the predictions according to the latest data monitoring.).	Independent service, running regardless of status change in energetic or environmental indicators.	The results will empower decision maker for efficient operation/manoeuvres.
UC01T04	UC1	Smart meter performance	ZIV	Continuous execution. Different readings scheduled at different intervals.	Meters registered in the HES	Billing data and other information available in the HES
UC01T05	UC1	Feeder mapping detection	ZIV	Continued execution once launched. On-demand	LV network topology manually recorded or resolved in previous iterations	LV network topology resolved with data added from advanced supervision and smart metering infrastructures
UC01T06	UC1	Web services functionality	OPA	Periodic execution once launched.	Network operating normally, devices	Obtaining the updated information of the

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
				On-demand	connected with MV RTU, and communications with central server active.	Secondary Substation devices: instantaneous measurement values, log of historical and events, as well as inventory of the equipment and its operating versions.  MV RTU functionality update
UC02T01	UC2	Energy box performance	CIRCE	Reading timer / polling request	Energy Box connected and operating (FUSE server connection)	Update read values and upload to the FUSE server (timestamp comparison)
UC02T02	UC2	ZIV feeder protection relay performance	ZIV	Fault in the protected feeder or in another feeder (the oscillography should be set to be triggered not only by the trip of the protection functions but also by their pick-up)	Network operating normally and feeders connected to the primary substation energized	Trip the breaker of the protected feeder if the fault is internal. Reclose the breaker if reclosing was enabled and the reclosing conditions are fulfilled. Not tripping if the fault is external

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC02T03	UC2	ZIV fault passage indicator performance	ZIV	Fault in MV network, in the same feeder and substation busbar where the system is connected, either downstream or upstream the connection point	Network operating normally and feeders feeding the secondary substation under test energized	Fault detector reporting faults downstream the connection point, and not reporting faults upstream the connection point or in different feeders
UC02T04	UC2	CIRCE fault locator training	CIRCE	Fault signal from protection IED (Simulated - External)	Simulated network with normal operating parameters. Capture of pre-fault network impedance data.	Simulated network with fault event, protective circuit breaker opening, and zone isolation performed. Capture of post-fault network impedance data. Feed the data set for training the model.
UC02T05	UC2	CIRCE fault locator verification	CIRCE	Fault signal from protection IED (real from field - External)	Spanish network with normal operating parameters. Capture of pre-fault network impedance data.	Spanish network with fault event, protective circuit breaker opening, and zone isolation performed. Capture of post-fault network impedance data. Determine the distance (calculated from the model)

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
						and compare with the real distance of the fault.
UC02T06	UC2	Flexibility test algorithm (optimal operation + emergency)	CIRCE	Network status discriminator request (1 minute for contingencies or 24 hours for optimal operation)	Simulated network with operating parameters based on estimated behaviour (load and generation), obtaining cable loads and network voltage profiles.	Determine flexibility request based on network status, send the set point to the respective assets (if necessary)
UC02T07	UC2	Self-Healing test algorithm	CIRCE	Fault event from the simulated network	Simulated network with normal operating parameters.	Simulated network with fault event, protective circuit breaker opening, and zone isolation performed. Fault section determination and sending of the reclosing sequence (simulated breakers and Energy Box in field)

The following sequence diagrams (see Figure 63 and Figure 64) reflect what will happen in Use Cases 1 and 2.

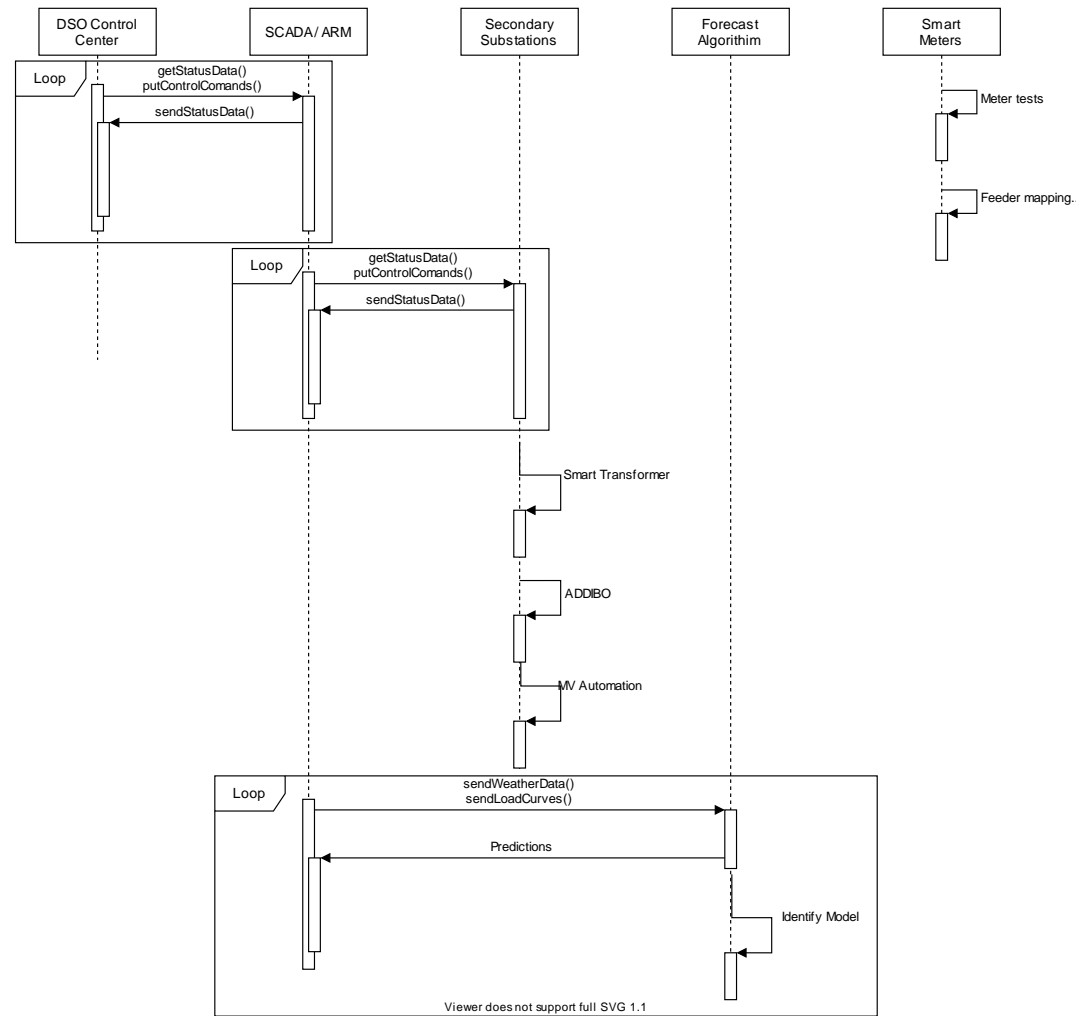
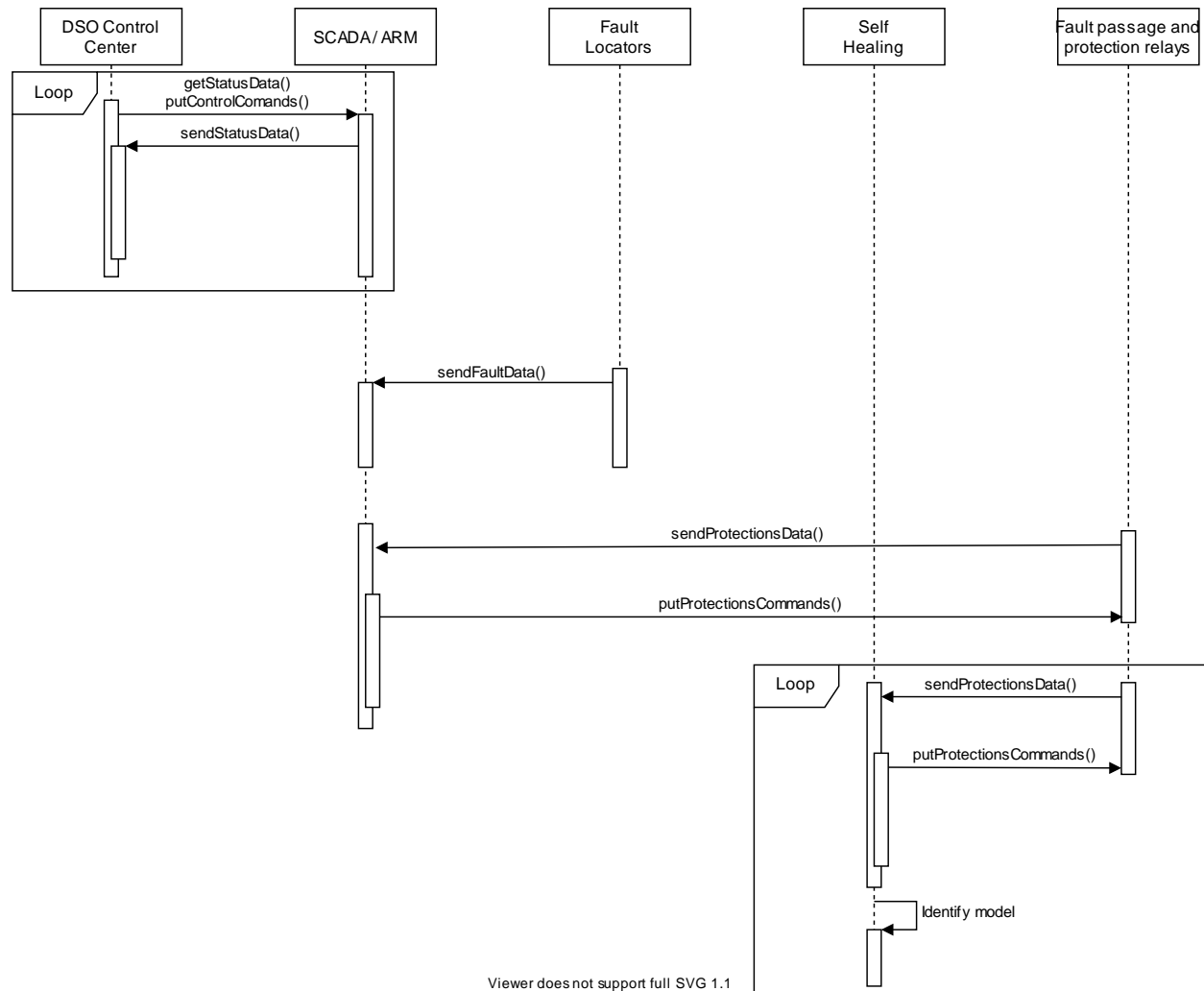


Figure 63. Spanish Use Case 1 Sequence Diagram (v0)



Viewer does not support full SVG 1.1

Figure 64. Spanish Use Case 2 Sequence Diagram

### Greek Demonstrator

This demonstrator will include use cases 3 and 4 and their related trials. Table 5 introduces them all and provides a glimpse on such relevant topics as what will trigger the event that causes that trial and the associated pre and post conditions.

In addition, Figure 65 presents in the same Sequence Diagram both uses cases.

Table 5: Scenarios of Greek Demonstrator

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC03T03	UC3	Validation of cost reduction benefits	VERD	<p>The forecast and scheduling optimisation modules trigger the battery set points</p> <p>Cost reduction estimation will be performed periodically upon request from VERD's side</p>	<p>Establishment of the necessary data streams.</p> <p>Communication between the Energy Box and FUSE platform</p>	Battery setpoint applied for the duration of the trial period and the cost reduction benefits are calculated based on the usage of the PV and battery systems
UC04T05	UC4	Validation of the reduction of network charges	VERD	<p>The forecast and scheduling optimisation modules trigger the battery set points</p> <p>Cost reduction estimation will be performed periodically upon request from VERD's side</p>	<p>Establishment of the necessary data streams.</p> <p>Communication between the Energy Box and FUSE platform</p>	Battery setpoint applied for the duration of the trial period and the network charges are calculated based on the usage of the PV and battery systems and compared to BAU charges

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC03T06	UC3	Black-out support (Simulation of islanded operation)	VERD	The microgrid congestion management module triggers the battery set points when a black-out is detected	Establishment of the necessary data streams.  Communication between the Energy Box and FUSE platform	Battery setpoint applied for the duration of the trial and the corresponding KPIs on black-out support are calculated
UC04T07	UC4	Peak shaving operation (active power support)	VERD	The microgrid congestion management module triggers the battery set points for peak shaving operation	Establishment of the necessary data streams.  Availability of the optimisation algorithms  Communication between the Energy Box and FUSE platform	Battery setpoint applied for the duration of the trial period and the peak reduction in the peak shaving efficiency is calculated utilising only active power dispatch from the inverters
UC04T08	UC4	Peak shaving (Reactive power support)	VERD	The microgrid congestion management module triggers the battery set points for peak shaving operation	Establishment of the necessary data streams.  Availability of the optimisation algorithms  Communication between the Energy Box and FUSE platform	Battery setpoint applied for the duration of the trial period and the peak reduction in the peak shaving efficiency is calculated utilising only reactive power dispatch from the inverters

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC04T09	UC4	Simulation of network congestion management (Demand Response)	VERD	The microgrid congestion management module triggers the battery set points for demand response operation	Establishment of the necessary data streams. Availability of the optimisation algorithms Communication between the Energy Box and FUSE platform	Battery setpoint applied for the duration of the trial period and the load demand of the whole substation is calculated to identify the demand response efficiency

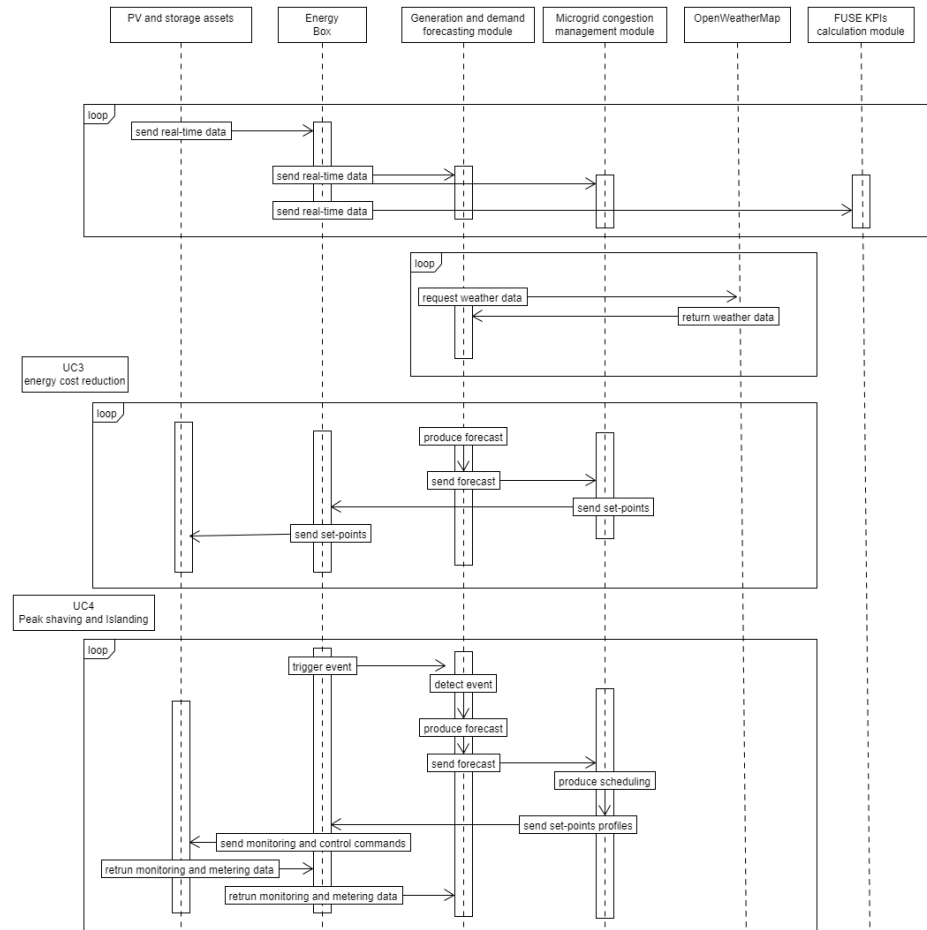


Figure 65. Greek Use Cases 3 & 4 Sequence Diagram

### Croatian Demonstrator

Concerning UC6, Virtual Energy Storage for Urban Buildings, one trial has been described for the operational phase. Other trials, performed during commissioning, will be undertaken in order to evaluate sub-sequences on the operational trial. As such, only one trial and corresponding sequence diagram are described below, which cover the other activities of the UC as well.

Table 6: Scenarios of Croatian Demonstrator

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC06T05	Virtual Energy Storage for Urban Buildings	Participation of VES assets on peak demand reduction and/or voltage and current congestion management scenarios through provision of flexibility	HYPERTECH	<p>The data retrieval and storage functionality are performed continuously.</p> <p>The model fitting operations are initiated automatically and are scheduled to be performed on a periodic basis.</p> <p>Flexibility estimation may be performed periodically or upon request from the DSO-side central optimizer.</p> <p>The asset control functionalities are triggered upon receipt</p>	<p>Establishment of necessary data streams. Available thermal models and optimization engine.</p> <p>Communication between DSO side and VES, as well as VES and the building(s)</p>	DR request is generated by the DSO, taking into account the available flexibility, and the VES coordinates the asset operation in order to satisfy the request.

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
----	---------------	------------	---------------	------------------	---------------	----------------

of a DR request by the  
DSO-side central  
optimizer.

The sequence diagram for UC06T05 is shown in Figure 66 below.

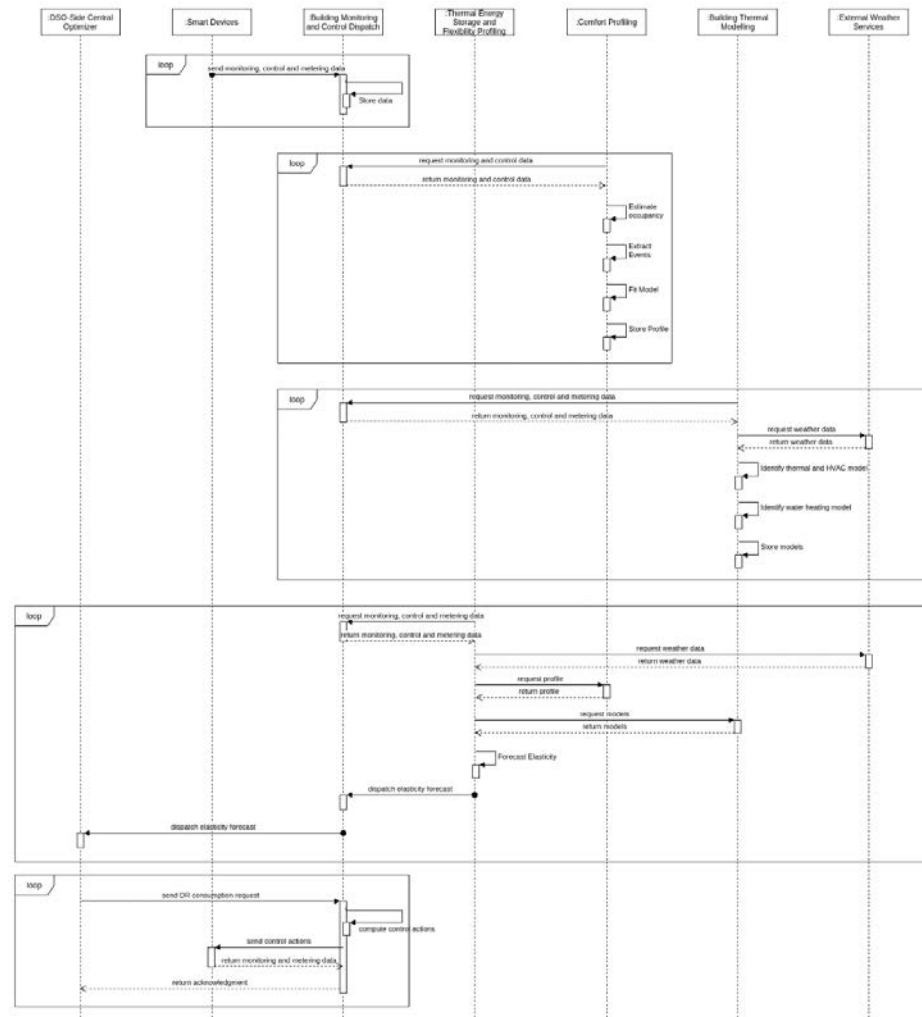


Figure 66. Croatian Use Case Sequence Diagram

### Italian Demonstrator

This demonstrator will include use cases 7 and 8 and their related trials. Table 7 introduces them all and provides a glimpse on such relevant topics as what will trigger the event that causes that trial and the associated pre and post conditions.

*Table 7: Scenarios of Italian Demonstrator*

ID	Use Case Name	Trial Name	Primary Actor	Triggering Event	Pre-Condition	Post-Condition
UC07T01	UC7	Master Network Loader	SELTA/LINKS	The DSO updates the .xml file that includes the topological and electrical changes about the MT grid or the SCADA configuration	Automatic software of SGC that reads the .xml file and updates the grid static data	All the static data are updated within the Static Network Database
UC07T02	UC7	Scenario Loader	SELTA/LINKS	Cyclic period (4 seconds)	OPC UA interface between SCADA and SGC	All the real time data (states and measures are setting on the SGC Dynamic Network Database
UC07T03	UC7	State estimator / Real time Load Flow solver	SELTA/LINKS	Cyclic period (4 seconds)	To ensure that T01 and T02 are completed	Runtime load flow calculation and basic evaluation about electrical values of the nodes and lines of the grid
UC07T04	UC7	Smart Grid Controller Core	SELTA/LINKS	Cyclic period (4 seconds): applied only	No ongoing grid violations	The system runs output for Island mode (UC8), Q/V

				for the user-selected modality	regulation or P regulation according to DSO setpoints or automatic control loop	
UC07T05	UC7	Set Point manager	SELTA/LINKS	Setpoint generated by Smart Grid Controller Core	OPC UA interface between SCADA and SGC	SCADA receives commands and setpoints for the Distributed Energy Resources
UC08T01	UC8	Island mode 'check and try'	SELTA/LINKS	Island selection by the DSO user	No ongoing grid violations, no ongoing regulation modes, power flow calculations completed	DSO user obtains recommended actions in order to manage the MV grid portion operating in islanding mode

The sequence diagram for UC7 and UC8 are shown in Figure 67 below.

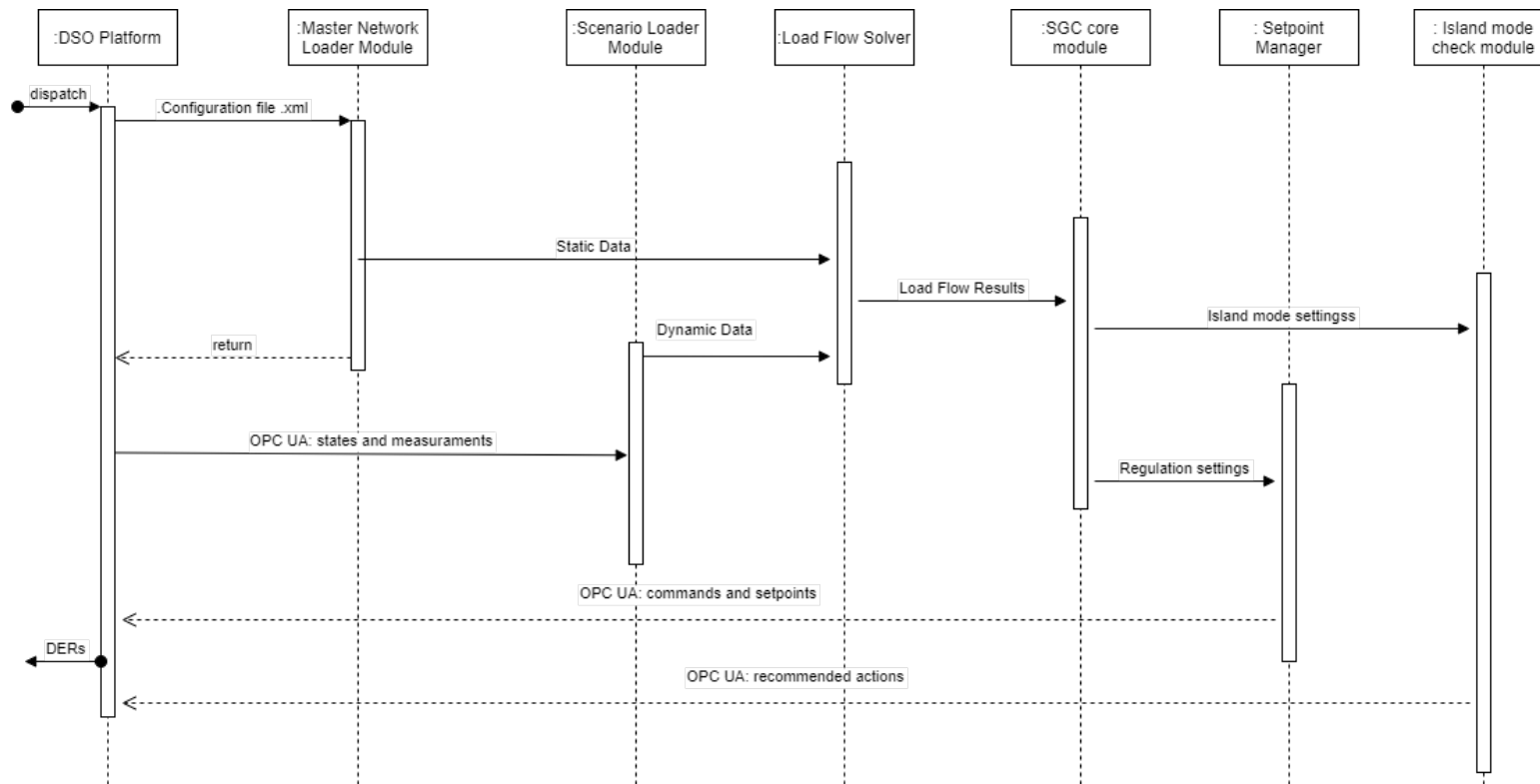


Figure 67. Italian Use Case Sequence Diagram

## 12. ANNEX 4 – FLEXIGRID scenarios view

### 12.1. Functional Requirements

The following specification of requirements is based on the information gathered from diverse activities and reflected in distinct project deliverables. What this chapter describes now are the requirements including the relevant information related to the FLEXIGRID Framework architecture.

The Functional requirements are presented in a particular format such as the one below, where the Requirement ID has the following syntax: Use Case Id / Unique Id Requirement within the Use Case

Requirement ID	X/Y
Author	
Description	
Type	Functional
Date	DD/MM/YYYY
Rationale	
Use case	UCX: Name
Fit & acceptance criteria	
Priority	Critical / Non-critical
Dependencies/Conflicts	
Comments	

Now, a list of functional requirements from the Spanish, Greek, Croatian and Italian demonstrators follow, as well as those ones that may affect more than one of them or them all (in which case 'Use Case Id' equals 0).

Requirement ID	0/1
Author	ATOS
Description	The developed solutions should be able to deal with diverse data formats.
Type	Functional
Date	29/07/2021
Rationale	The nature of project includes working with a few sorts of data such as time data, power flows, etc.
Use case	UC0: General requirements
Fit & acceptance criteria	The system shows its adaptors make it capable to perform the expected activities with different data formats.
Priority	Critical / Non-critical
Dependencies/Conflicts	--
Comments	--

Requirement ID	0/2
Author	ATOS
Description	Enough data storage capacity should be available

Type	Functional
Date	29/07/2021
Rationale	Project should have enough data storage capacity
Use case	UC0: General requirements
Fit & acceptance criteria	Check all the data can be handled by the system
Priority	Critical
Dependencies/Conflicts	--
Comments	--

Requirement ID	<b>0/3</b>
Author	ATOS
Description	Periodic evaluation of data quality should be performed to manage different sources of erroneous data.
Type	Functional
Date	29/07/2021
Rationale	Because of the amount of data produced, occasional checks should be given to stay away of future issues.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking periodic data control are established.
Priority	Medium
Dependencies/Conflicts	--
Comments	--

Requirement ID	<b>0/4</b>
Author	VERD
Description	Units should be concise within FUSE based on KPIs description
Type	Functional
Date	24/08/2021
Rationale	Units should be concise within FUSE to avoid errors in calculations and for clear understanding of the available datasets
Use case	UC0: General requirements
Fit & acceptance criteria	The units of the uploaded data streams are aligned to what is described in the KPIs definition tables in D6.1
Priority	Medium
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>0/5</b>
Author	VERD
Description	FUSE platform should be able to calculate all KPIs defined In D6.1 for different time periods defined by the user
Type	Functional
Date	23/08/2021
Rationale	Pilot managers need to be able to report the trials' outcomes based on the KPIs calculated by FUSE using raw data from the pilots

Use case	UC0: General requirements
Fit & acceptance criteria	Pilot managers to be able to define time periods and KPIs and download them in an csv file
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>0/6</b>
Author	VERD
Description	FUSE platform should be able to allow pilot managers to define time periods and choose raw data sets to be downloaded in csv files
Type	Functional
Date	23/08/2021
Rationale	Pilot managers need to be able to analyse the trial data
Use case	UC0: General requirements
Fit & acceptance criteria	Pilot managers to be able to define time periods and raw data sets to be downloaded in csv files
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>0/7</b>
Author	VERD
Description	FUSE platform should be able show to pilot managers whether all assets are fully functional and transmitting data as per their settings
Type	Functional
Date	23/08/2021
Rationale	Pilot managers need to be able to see if there are any issues with their assets transmitting data
Use case	UC0: General requirements
Fit & acceptance criteria	Pilot managers to be able to evaluate the status of their assets
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>0/8</b>
Author	VERD
Description	FUSE platform should be able to receive data points from the assets
Type	Functional
Date	23/08/2021
Rationale	Data should be managed and stored within FUSE platform
Use case	UC0: General requirements
Fit & acceptance criteria	Established communication between FUSE platform and all pilot sites' assets for the data exchange process

Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>3/1</b>
Author	VERD
Description	FUSE platform should be able to send set points commands to the Energy Box to be applied to the field devices
Type	Functional
Date	23/08/2021
Rationale	Established communication between the Energy Box and FUSE platform is essential not only for data exchange but also for commands that need to be applied to the onsite assets through the Energy Box
Use case	UC3 and UC4
Fit & acceptance criteria	Established communication between FUSE platform and all pilot sites' assets for the data exchange process
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>3/2</b>
Author	VERD
Description	FUSE platform should be able to store the PV forecast and the load forecast from the forecasting module (ER6) and allow the user to download the forecasts
Type	Functional
Date	23/08/2021
Rationale	The users should be able to view and download the results of the assessments of the module in order to evaluate its operation
Use case	UC3
Fit & acceptance criteria	Results downloaded and stored in FUSE
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>4/1</b>
Author	VERD
Description	FUSE platform should be able to store the results of the scheduling module (ER7) and allow the user to download the results
Type	Functional
Date	23/08/2021
Rationale	The users should be able to view and download the results of the assessments of the module in order to evaluate its operation

Use case	UC4
Fit & acceptance criteria	Results downloaded and stored in FUSE
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>6/1</b>
Author	Hypertech
Description	Establishment of smart data acquisition and automated energy management
Type	Functional
Date	18/08/2021
Rationale	UC6 depends on the availability of real-time monitoring and metering data and the ability to establish an automated control framework
Use case	UC6: Name
Fit & acceptance criteria	Monitoring data are communicated from the pilot site to Hypertech's module. Remote control actions are successfully sent by the module to the HVAC devices.
Priority	Critical
Dependencies/Conflicts	-
Comments	-

Requirement ID	<b>6/2</b>
Author	Hypertech
Description	Ability to fit models for comfort profiling
Type	Functional
Date	18/08/2021
Rationale	Flexibility estimation must take into consideration the preferences of the users, as extracted from recorded data.
Use case	UC6: Name
Fit & acceptance criteria	Enough data are generated in order to be able to train the profiling model
Priority	Critical
Dependencies/Conflicts	6/1
Comments	-

Requirement ID	<b>6/3</b>
Author	Hypertech
Description	Ability to identify thermal models for the devices
Type	Functional
Date	18/08/2021
Rationale	Flexibility estimation and device control require the ability to mathematically model the devices and identify the model parameters based on measured data.
Use case	UC6: Name

Fit & acceptance criteria	Enough data are generated in order to be able to train the thermal mode. Validation error less than 25%.
Priority	Critical
Dependencies/Conflicts	6/1
Comments	-

Requirement ID	<b>6/4</b>
Author	Hypertech
Description	Accurate estimation of flexibility and ability to implement a DR request
Type	Functional
Date	18/08/2021
Rationale	In order to satisfy a DR request, the system must accurately predict baseline and available flexibility, and be able to apply the control signals to follow a requested demand curve.
Use case	UC6: Name
Fit & acceptance criteria	Virtual Energy Storage for Urban Buildings
Priority	Critical
Dependencies/Conflicts	6/1
Comments	-

Requirement ID	<b>1/1</b>
Author	UNICAN
Description	Self-managed transformer for Secondary Substation
Type	Functional
Date	19/08/2021
Rationale	The transformer must be capable to on load self-commutation
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Interoperability with SCADA and ARM Systems
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>1/2</b>
Author	UNICAN
Description	Equipment integration through Energy Box
Type	Functional
Date	19/08/2021
Rationale	Integration of different equipment among them and with FUSE
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Data received from assets and connection to FUSE
Priority	Secondary (for Spanish Demo Site)
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>1/3</b>
Author	UNICAN
Description	Supervision of LV side on Secondary Substations
Type	Functional
Date	19/08/2021
Rationale	Interoperability with SCADA and ARM Systems
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Data received from assets and connection to FUSE
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>1/4</b>
Author	UNICAN
Description	Forecasting algorithm
Type	Functional
Date	19/08/2021
Rationale	Prediction of energy generation from renewable sources
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Error margin between -10% and +10% deviation of forecasted and real data
Priority	Secondary (for Spanish Demo Site)
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>2/1</b>
Author	UNICAN
Description	ZIV protections
Type	Functional
Date	19/08/2021
Rationale	Performance of protection relays and fault passage indicators
Use case	UC2: Protections functions operating with large RES share penetration in the distribution grid
Fit & acceptance criteria	No false trips and no missed trips
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>2/2</b>
Author	UNICAN
Description	CIRCE fault locators
Type	Functional
Date	19/08/2021

Rationale	Performance of new reflectometry fault locators
Use case	UC2: Protections functions operating with large RES share penetration in the distribution grid
Fit & acceptance criteria	The system determines the distance within an error range that effectively locates the fault
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>1/5</b>
Author	UNICAN
Description	Feeder mapping detection
Type	Functional
Date	19/08/2021
Rationale	The feeder mapping algorithm should determine the feeder and phase of all the meters fed by the secondary substation
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Over 85% of feeders correctly mapped
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>1/6</b>
Author	UNICAN
Description	Web Services functionality
Type	Functional
Date	19/08/2021
Rationale	Web services functionality for RTU equipment at Secondary Substations
Use case	UC1: Secondary Substation upgrading for higher grid automation and control
Fit & acceptance criteria	Interoperability with SCADA and ARM Systems
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>2/3</b>
Author	UNICAN
Description	Flexibility test algorithm (optimal operation + emergency)
Type	Functional
Date	19/08/2021
Rationale	It allows controlling the assets in the network, adjusting its operation to present and future conditions.
Use case	UC2: Protections functions operating with large RES share penetration in the distribution grid

Fit & acceptance criteria	Provides effective settings to avoid network congestion and optimally configures network performance under normal operating conditions
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>2/4</b>
Author	UNICAN
Description	Self-Healing test algorithm
Type	Functional
Date	19/08/2021
Rationale	Detects the faulted section of the network and sends the appropriate closing command to restore the service
Use case	UC2: Protections functions operating with large RES share penetration in the distribution grid
Fit & acceptance criteria	Consistently detects faulty sections and reduces network outage time
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>7/1</b>
Author	EDYNA
Description	Automatic detection of the data grid from SCADA system
Type	Functional
Date	28/09/2021
Rationale	The system must be able to load all the data and arrangement of the medium voltage grid from EDYNA SCADA automatically
Use case	UC7: Dispatching platform for MV generation
Fit & acceptance criteria	The grid model loaded in the SGC (Smart Grid Controller) is the same of the SCADA system
Priority	Critical
Dependencies/Conflicts	
Comments	-

Requirement ID	<b>7/2</b>
Author	EDYNA
Description	Dispatching platform
Type	Functional
Date	28/09/2021
Rationale	The SGC platform acquires the whole measures and information for the dispatching platform
Use case	UC7: Dispatching platform for MV generation
Fit & acceptance criteria	The dispatching platform works correctly, without voltage violations and/or congestions
Priority	Critical

Dependencies/Conflicts	7/1
Comments	-

Requirement ID	8/1
Author	EDYNA
Description	Grid in island mode
Type	Functional
Date	28/09/2021
Rationale	The SGC system maintains the grid in island mode
Use case	UC8: Mountainous valley grid operating in island mode
Fit & acceptance criteria	The grid in island mode is stable, with frequency and voltage inside the limits imposed by the rules
Priority	Critical
Dependencies/Conflicts	7/1
Comments	-

## 12.2. Non-Functional Requirements

The non-functional requirements posed by the FLEXIGRID platform are summarized below, assorted and introduced through some summaries rather similar as the ones employed to present the functional requirements.

Requirement ID	<b>X/Y</b>
Author	Partner in the consortium providing this requirement
Description	Sentence describing the requirement
Type	Non-Functional
Date	DD/MM/YYYY
Rationale	Explain this requirement
Use case	UCX: Name
Fit & acceptance criteria	How to check FLEXIGRID complies with this requirement
Priority	Critical / Non-critical
Dependencies/Conflicts	Links to other requirements
Comments	Any additional observation

The final list of non-functional requirements is as follows.

Requirement ID	<b>0/1</b>
Author	ATOS
Description	The developed solutions must be founded on open standards.
Type	Non-Functional
Date	29/07/2021
Rationale	Standards give individuals and organizations a premise for shared agreement
Use case	UC0: General requirements
Fit & acceptance criteria	Checking utilization of selected standards
Priority	Critical
Dependencies/Conflicts	Regulatory
Comments	

Requirement ID	<b>0/2</b>
Author	ATOS
Description	The developed solutions should be interoperable to guarantee high replication potential.
Type	Non-Functional
Date	29/07/2021
Rationale	European projects target researching to be accessible for sharing and public use, facilitating the practice and knowledge acquired for other projects.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking use of interoperability standards
Priority	Critical
Dependencies/Conflicts	
Comments	

Requirement ID	<b>0/3</b>
Author	ATOS
Description	The developed solutions should be scalable in terms of computation and communication.
Type	Non-Functional
Date	29/07/2021
Rationale	European projects effort should be implemented in a optimal way, hence allowing its developed components used in other projects with the least required possible effort and different sizes.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking modules adapt for different size of data and projects with the least effort
Priority	Critical
Dependencies/Conflicts	
Comments	

Requirement ID	<b>0/4</b>
Author	ATOS
Description	The developed solutions should not endanger the security of supply and reliability of the underlying electricity grid.
Type	Non-Functional
Date	13/09/2021
Rationale	The project cannot put at risk the pre-project conditions, always having improvement in mind.
Use case	UC0: General requirements
Fit & acceptance criteria	Solution implemented will not have access to underlying electricity grid unless a security system is implemented.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/5</b>
Author	ATOS
Description	The developed solutions should be cost efficient, deploying the least-cost technological alternatives.
Type	Non-Functional
Date	13/09/2021
Rationale	Cost efficiency must be a common standard so long as complying with the objectives.
Use case	UC0: General requirements
Fit & acceptance criteria	Cost efficiency must be implemented in all aspects of the project, e.g. use of open source software.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/6</b>
Author	ATOS
Description	Involved data exchanges should not be prone to security hazards.
Type	Non-Functional
Date	13/09/2021
Rationale	Working with personal data requires taking into account security hazards.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking use of tools with security approach incorporated as Block-Chain technology.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/7</b>
Author	ATOS
Description	Communication systems must be based on a specific and pre-defined set of protocols.
Type	Non-Functional
Date	13/09/2021
Rationale	Main part in integration different modules rely on protocols. Such protocols must be preestablished to work in an efficient manner and save future problems.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking different protocols communications among modules before starting modules' development.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/8</b>
Author	ATOS
Description	Supporting documentation should be translated to different languages.
Type	Non-Functional
Date	13/09/2021
Rationale	The European union end users and stake holders should have access to information
Use case	UC0: General requirements
Fit & acceptance criteria	Checking relevant documentation has been translated to different languages.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/9</b>
Author	ATOS

Description	The developed solutions should conform to the current market and regulatory framework in the pilot countries and the EU in general.
Type	Non-Functional
Date	13/09/2021
Rationale	In order to be implemented and have a general use the project must comply with current market and regulatory framework.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking solutions comply with current market and regulatory framework.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/10</b>
Author	ATOS
Description	The developed solutions should be adaptable to potential future changes in the EU market and regulatory framework.
Type	Non-Functional
Date	13/09/2021
Rationale	The solution must be available for a long period of time therefore be adaptable to EU changes in market and regulatory framework.
Use case	UC0: General requirements
Fit & acceptance criteria	Checking parts subjected to regulations must be traced and accessible.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/11</b>
Author	ATOS
Description	The data management platform should be designed according to the principles of modularity, scalability, and interoperability.
Type	Non-Functional
Date	13/09/2021
Rationale	In order for the project to be used in different scenarios, having the possibility for incorporate and eliminate modules in a flexible manner and adapt to different size of data handling.
Use case	All
Fit & acceptance criteria	Checking data management platform work with modules are well defined, and tools selected allow to work with different amounts of data.
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/12</b>
Author	ATOS
Description	Incorporation of as much as possible open standards in order to ensure the development of standardized framework.
Type	Non-Functional
Date	13/09/2021
Rationale	Standards provide people and organizations with a basis for mutual understanding
Use case	All
Fit & acceptance criteria	Checking use of selected standards
Priority	Critical
Dependencies/Conflicts	

Requirement ID	<b>0/13</b>
Author	ATOS
Description	The data management platform should be able to promote new standards if the existing ones are not covering the project needs.
Type	Non-Functional
Date	13/09/2021
Rationale	Standards provide people and organizations with a basis for mutual understanding, if fields involved do not possess suitable standards new ones must be implemented
Use case	All
Fit & acceptance criteria	Checking use of selected standards if any and create new ones with their own ontology
Priority	Critical
Dependencies/Conflicts	--

## 13. ANNEX 5 – FLEXIGRID CIM entities creation example

Below there is an example of an entity defined for the FLEXIGRID CIM that allows to illustrate the mapping of the standard CIM to a JSON file to allow integration with the FUSE platform and share it with the partners. This example helps to understand the Venn diagram shown in Section 3. It also shows how entities are referenced via IRI (Internationalized Resource Identifiers) with the underlying CIM model.

In this particular case, an entity called *PowerTransformer*, keeps the same existing name in the CIM standard. Before showing the content of the .JSON files defined for each entity, it is worth mentioning that a series of restrictions have been applied to said content to allow a correct integration with the FUSE platform.

In the following example code, some attributes defined for each entity are mandatory and the rest optional. The first entity defined is *PowerTransformer*, that is mandatory.

Despite the fact that the *PowerTransformer* entity exists as such in the CIM standard, a series of restrictions have been applied when defining the content of its corresponding JSON file, in order to be able to be integrated into FUSE. The idea behind this approach is that, when this JSON file is imported into a simulation software, the contexts of both FIWARE and the used simulation software are well known and connected.

The main features that define the content of any of the existing entities in the FLEXIGRID CIM are highlighted as comments in the .JSON file content.

Table 8 (Code): Example of PowerTransformer entity

```
flexigrid_cim-power_transformer_example.json

{
  "id": "spain:grid:...:toranzo"
  # <highest_hierarchy_level_component_id>: <other_level_components_id>: <lowest_level_component_id>

  "type": "PowerTransformer",

  # Standard Attributes

  "dateCreated": {
    "type": "ISO8601",
    "value": "2021-01-13T08:15:26.492Z",
    "metadata": {}
  },
  "dateModified": {
    "type": "ISO8601",
    "value": "2021-01-13T08:15:26.492Z",
    "metadata": {}
  },

  # Optional Attributes

  "name": { # Name of entity **after** being included in a simulation software and whenever is
    used an entity is used in another context

    "type": "StructuredValue",
    "value": [ # list of contexts where this entity can be found

      "GridModel:DigSilent:http://example.org/semantic_repository/spain/toranzo/19700101T0000Z_YYY_EQ.xml#_d14e0791-30f1-4d5b-9119-e832baf8e7d7",
      # The IRI identifier_d14e0791-30f1-4d5b-9119-e832baf8e7d7 comes from DigSilent.
      The process of how to locate this identifier in the .xml generated by DigSilent is explained in
      the Annex.

      "GridLocationPrimary:spain:Toranzo:http://example.org/semantic_repository/spain/toranzo/19700101T0000Z_YYY_TP.xml#_0cd0fe1a-45bb-f7c4-aabc-edd895cf8774", #grid_location_hv_id",
      "GridLocationPrimary:spain:Toranzo:http://example.org/semantic_repository/spain/toranzo/19700101T0000Z_YYY_TP.xml#78521439-95d9-3f6b-2818-0d6687b69e6c" #grid_location_mv_id

    ],
    "metadata": {}
  },
  "description": {
    "type": "Text",
```

```

    "value": "< Transformador del CT de Toranzo>", # Description of this specific instance of
the entity

    "metadata": {}

},

# Mandatory Attributes

"category": {
    "type": "Text",
    "value": "<hv|mv>",
    "metadata": {}
},

"PrimaryThreePhaseVoltage": {
    "type": "StructuredValue",
    "value": {
        "L1": 55000,
        "L2": 55000,
        "L3": 55000
    },
    "metadata": {
        "timestamp": {
            "type": "DateTime",
            "value": "2020-12-21T13:16:00.173Z"
        },
        "isMeasuredIn": {
            "type": "Text",
            "value": "volt"
        },
        "aggregatedType": { # measurementType also works for us
            "type": "Text",
            "value": "rms"
        },
        "measurementInterval": { # Sample frequency
            "type": "Number",
            "value": 1
        }
    }
},

```

```

    "SecondaryThreePhaseVoltage": {
      "type": "StructuredValue",
      "value": {
        "L1": 55000,
        "L2": 55000,
        "L3": 55000
      },
      "metadata": {
        "timestamp": {
          "type": "DateTime",
          "value": "2020-12-21T13:16:00.173Z"
        },
        "isMeasuredIn": {
          "type": "Text",
          "value": "volt"
        },
        "aggregatedType": { # measurementType also works for us
          "type": "Text",
          "value": "rms"
        },
        "measurementInterval": { # Sample frequency
          "type": "Number",
          "value": 1
        }
      }
    }
  }
}

```

The FUSE context broker, as of the delivery date of this document, already supports NGSI-LD although FLEXIGRID CIM uses the previous version (NGSI v2).

The use of identifiers in this way is motivated by the work in DLV 5.3 *“Protocols and standards definition”*, where it was explained that using hierarchical models helps the execution of certain tasks.

As detailed as the work methodology to be used is defined above, not all entities needed in the frame of FLEXIGRID project described earlier have representation in the CIM standard. Those entities need to be represented in JSON format to exchange information among modules.

## 14. ANNEX 6 - Cybersecurity in the 4 areas of interest in a smart grid

Here readers may find a summary of the cybersecurity measurements associated to each one of the four areas identified in FLEXIGRID's DoA.

### 14.1. Equipment Security

An electricity grid supplies power that originates from a plant of electricity generation to the consumers. This electricity can be generated from either renewable (solar, wind, and so on) or non-renewable sources (coal and diesel). The main components of the electricity grid are the high voltage transmission lines that connect the power plant to the transmission substation, and the low/medium voltage distribution lines that connect the substations to neighbourhoods and ultimately consumers [58].

Energy systems combine legacy equipment, in some cases installed decades ago and not prepared to deal with cybersecurity, with state-of-the-art new digital equipment following the security-by-design principle, but commonly exposing some of the legacy equipment to unforeseen digital threats. In addition, the Internet of Things (IoT) devices' incorporation into energy systems leads to additional risk. Most of these devices are not compliant with the strict requirements for the security of energy networks and there is a high risk of malicious usage if connecting them with no security or trust assurance. One of the hot security topics discussed when addressing Confidentiality, Integrity, Availability (CIA) security is how vulnerable current supply chains are. The risk that the supply chain for electronic components or ICT technologies, including microchips, embedded software, SCADA and control applications, operating systems, etc. could be infiltrated at some stage by hostile agents is very real. These hostile agents could alter the circuitry of the electronic components or substitute counterfeit components with altered circuitry. Moreover, backdoors and logic bombs and other malicious software could be included as part of the firmware of many Intelligent Electronic Devices (IEDs), controllers, or smart meters. As a result, enemy states, terrorists, or any other threat could make use of a backdoor to get remote control of the affected information systems or alternatively take advantage of preinstalled logic bombs that could cause terrible harm.

The security of the supply chain is of paramount importance for smart grids protection. This is especially true for those applications and components that could be relevant for national security. The design, fabrication, assembly, and distribution of the electronic components and applications will have to be controlled and appropriately regulated. It is important to have in mind the economical dimension of the problem and establish security objectives that are economically viable. The key to solving the problem of malicious firmware is to make the entire global supply chain more secure. Device protection is a crucial element in the supply chain of smart grid security. Many research papers and recommendation reports have been published contributing to security assurance for endpoints. Several security technologies have been recommended, particularly, host IDS, anti-virus, and host data loss prevention (DLP). Additionally, the recommendation includes an automated security compliance check. Such a tool performs checks against all smart grid components to verify that each device's configuration is up to date, especially the device's firmware and the current configuration file. As the smart



managed by FLEXIGRID has been analysed to identify the best approach to securely manage the data produced by the four different pilots. By following the General Data Protection Regulation (GDPR) regulation, data that refers to any information relating to an identified or identifiable natural person should be anonymized. The typical techniques to prevent the exposure of personal data are:

- **Masking:** this approach involves allowing access to a modified version of sensitive data. This can be achieved by modifying data in real time, as it is accessed (dynamic data masking) or by creating a mirror version of the database with anonymized data (static data masking) through encryption, term or character shuffling, or dictionary substitution.
- **Generalization:** this approach requires excluding certain data to make it less identifiable. Data could be changed into a range of values with logical boundaries.
- **Swapping:** this approach also called shuffling or data permutation, rearranges dataset attribute values so that they do not match the initial information.
- **Perturbation:** changes the initial dataset slightly by using rounding methods and random noise. The values used must be proportional to the disturbance employed.

Since the information exchanged within FLEXIGRID does not involve any Personal Identifiable Information (PII), it was not necessary to apply any kind of anonymization. Furthermore, encryption at the database level has been avoided to reduce as much as possible the impact of the computational delays of the cryptographic functions. The data collected on the cloud storage and exchanged between pilots and remote services has been protected through the dedicated security credentials based on the OAuth2 standard later described.

#### 14.4. Platform Integration Security

The first step of a secure integration among the set of distributed software components built in FLEXIGRID refers to the production of clear documentation for the application interfaces. All the public REST-based APIs offered by the cloud platform are defined and documented via OpenAPI specifications.

The OpenAPI Specification (OAS) implemented through swagger [60], defines a standard, language-agnostic interface to RESTful APIs which allows both humans and computers to discover and understand the capabilities of the service without access to source code, documentation, or through network traffic inspection. When properly defined, a consumer can understand and interact with the remote service with a minimal amount of implementation logic. An OpenAPI definition can then be used by documentation generation tools to display the API, code generation tools to generate servers and clients in various programming languages, testing tools, and many other use cases. OpenAPI v2.0 was previously known as Swagger before being donated to the OpenAPI Initiative. In contrast to JSON Schema, an OpenAPI document is a definition for an entire API, not just data models. Before its creation, many APIs were designed without any ability to map how they should work or validate that it operates as expected. With this machine-readable description, you can also generate useful tools for humans, such as documentation and mock servers. You can use JSON Schema to describe data objects for both requests and responses. However, OpenAPI includes how those requests and responses are formatted. The following picture shows a portion of the description produced to enable pilot and services secure integration, available at <https://unified-api.fuse.flexigrid-h2020.eu/docs>.

## FLEXIGRID 0.1.0 OAS3

/openapi.json

### default

GET	/pilot/{device}/{hits}/ Search Type	⌵
GET	/flexigrid_calculations/last/ Get Last Calculations	⌵
GET	/setpoints/battery/ Get Loads	⌵
POST	/setpoints/battery/ Post Setpoints	⌵
GET	/pilot/types/ Get Pilot Types	⌵

Figure 69. FUSE OpenAPI descriptions

The API built matches the requirement raised by the definition of the Common Interface Model (CIM) developed within Task 5.2. In particular, the necessity of data harmonization for the overall information flows conceived by FLEXIGRID led to the definition and development of the set of interfaces currently enabled by the FUSE platform. Successively, the Oauth2 open standard, in conjunction with OpenID-connect has been exploited to protect access to the cloud functionalities for the overall RESTful interfaces built. Additional information on the security standards exploited is further described in chapter 4.