PLATFORM FOR OPERATION OF DISTRIBUTION NETWORKS

Platone

PLATform for Operation of distribution NEtworks

D2.10

Specification of the interoperability and standard communication protocols (v2)

The project PLATform for Operation of distribution NEtworks (Platone) receives funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement no 864300.

Project name	Platone
Contractual delivery date:	31.08.2023
Actual delivery date:	30.08.2023
Main responsible:	RWTH Aachen University
Work package:	WP2 – Platform Implementation and Data Handling
Security:	P
Nature:	R
Version:	V1.0
Total number of pages:	27

Abstract

This deliverable defines the intra-platform interoperability mechanisms and standard communication protocols necessary for the Platone Open Framework implementation, extending the first version of this report with updates on the solutions aligned to specifications foreseen in the early stage of the project. In addition, it reports on proposed semantic solutions for enhancing data interoperability evaluating ontologies on payloads. It also presents how the solutions are implemented in the different components of the Platone Open Framework and field trials giving a focus on design and security best practices.

Keyword list

Platone Open Framework, Data Interoperability, Security Aspects, Communication protocols

Executive Summary

Within the H2020 programme “A single, smart European electricity grid”, Platone addresses the topic “Flexibility and retail market options for the distribution grid”. Modern power grids are moving away from centralised, infrastructure-heavy transmission system operators (TSOs) towards distribution system operators (DSOs) that are flexible and more capable of managing diverse renewable energy sources. DSOs require new ways of managing the increased number of producers, end users and more volatile power distribution systems of the future. The Platone Open Framework aims to create an open, flexible and secure system that enables distribution grid flexibility/congestion management mechanisms, through innovative energy market models involving all the possible actors at many levels (DSOs, TSOs, customers, aggregators). It is an open source framework that allows standard and flexible integration of external solutions (e.g. legacy solutions) in a secured manner, and is open to integration of external services through standardized open application program interfaces (APIs). It is built with existing regulations in mind and will allow small power producers to be easily certified so that they can sell excess energy back to the grid. Furthermore, it will also incorporate an open-market system to link with traditional TSOs.

The Platone platform solution consists of a layered architecture named Platone Open Framework. This deliverable focuses on interoperability, taking into account the selected communication protocols and data models for integrating the different platform components within the framework and with external systems. It presents semantic solutions by evaluating payloads with the use of ontologies to improve interoperability and explains the benefits of the semantic annotation and semantic similarity tools in this regard. In this context, relevant data models used in the project such as the standard CIM-61968-9 and the proprietary data model for phasor measurement unit measurements are taken into account. Furthermore, in term of standards, the IEC 62559 is another important standard that is used for the definition of use cases. Focusing on one of the core components of the Platone framework, i.e., the DSO Technical Platform (DSOTP), an overview of the relevant updates of the DSOTP services has been provided. This overview contemplates a redesign of the balancing service and the solutions that were given in term of design and communication with the Greek and German demos. Additionally, improvements have been made to the DSOTP Application Programming Interfaces (APIs) using the OpenAPI 3.0 standard. These improvements foster enhanced communication and compatibility by enabling seamless exploration and comprehension of the DSOTP services' capabilities. More specifically for the interfaces of DSOTP with external systems, two actions have been made. A shift in the data visualization from the DSOTP to the German demo Azure platform was pursued for the German demo while the development of converters needed to integrate the state estimation tool and a dashboard to represent the graph of the Greek demo network was conducted for the Greek demo.

Finally, the underlying communication protocols for intra-platform communication in terms of asynchronous and synchronous communication are presented. For communication between different components of the framework, both kinds of communication channels are secured with underlying transport layer security. Moreover, design principles and best practices with respect to security aspects are depicted, and an innovative mechanism for identity management and access control based on blockchain technology is presented.

Authors and Reviewers

Main responsible		
Partner	**Name**	**E-mail**
RWTH		
	Juan Galeano	adolfo.galeano@eonerc.rwth-aachen.de
Author(s)/contributor(s)		
Partner	**Name**	
ENG		
	Ferdinando Bosco	ferdinando.bosco@eng.it
RWTH		
	Juan Galeano	adolfo.galeano@eonerc.rwth-aachen.de
	Florian Oppermann	florian.oppermann@eonerc.rwth-aachen.de
Reviewer(s)		
Partner	**Name**	
NTUA		
	Paangiotis Pediaditis	panped@mail.ntua.gr
ARETI		
	Fedele Gabriele	Gabriele.Fedele@areti.it
Approver(s)		
Partner	**Name**	
RWTH		
	Amir Ahmadifar	amir.ahamdifar@eoenrc.rwth-aachen.de

Table of Contents

1 Introduction

The project "PLATform for Operation of distribution Networks – Platone" aims to develop an architecture for testing and implementing a data acquisition system based on a two-layer Blockchain approach: an "Access Layer" to connect customers to the Distribution System Operator (DSO) and a "Service Layer" to link customers and DSO to the Flexibility Market environment (Market Place, Aggregators, …). The two layers are linked by a Shared Customer Database, containing all the data certified by Blockchain and made available to all the relevant stakeholders of the two layers. The tools used for this purpose will be based on platforms able to receive data from different sources, such as weather forecasting systems or distributed smart devices spread all over the urban area. These platforms, by talking to each other and exchanging data, will allow collecting and elaborating information useful for DSOs, transmission system operators (TSOs), Market, customers and aggregators. In particular, the DSOs will invest in a standard, open, non-discriminatory, blockchain-based, economic dispute settlement infrastructure, to give to both the customers and to the aggregator the possibility to more easily become flexibility market players. This solution will allow the DSO to acquire a new role as a market enabler for end users and a smarter observer of the distribution network. By defining this innovative two-layer architecture, Platone strongly contributes to remove technical and economic barriers to the achievement of a carbon-free society by 2050 [1], creating the ecosystem for new market mechanisms for a rapid roll out among DSOs and for a large involvement of customers in the active management of grids and in the flexibility markets. The Platone platform will be tested in three European trials (Greece, Germany, and Italy).

As described above, the Platone platform solution consists of a layered architecture named Platone Open Framework. It mainly consists of the Blockchain Access Layer, handling data certification and flexibility activation based on smart contracts, the Platone Market Platform allowing for flexibility requests from TSOs and DSOs in a large geographical area and the Platone DSO Technical Platform that provides DSOs with a reference way to deploy auxiliary grid services. This Platone Open Framework architecture allows a greater stakeholder involvement and enables an efficient and smart network management and can be seen in Deliverable D2.1 [2].

To achieve smooth integration among all these components, it is crucial to align the interoperability and security of the different platform elements. This deliverable centres on examining the existing design principles of these components concerning data models, communication protocols, and security considerations. Moreover, it proposes semantic approaches that leverage ontologies to enhance interoperability and examines the benefits of employing semantic annotation and similarity tools in this context.

1.1 Task 2.4

Task 2.4 covers the intra-platform interoperability and standard communication protocols of the Platone Open Framework. This final version of the document focuses on the further development of the intra-platform communication between the three core components of the Platone Open Framework and external systems.

1.2 Objectives of the Work Reported in this Deliverable

As stated in the task description in the previous section, this final version of the deliverable provides an updated description of the intra-platform communication between the three core components of the Platone Open Framework (Blockchain Access Layer, DSO Technical Platform, and Market Platform) and further external systems.

The main goal of this report is to present the updates of the intra-platform communication in the Platone Open Framework from different points of view, in order to allow all stakeholders involved to understand the characteristics and consider possible integrations by providing the path that was taken within the platform.

1.3 Outline of the Deliverable

Chapter 2 introduces the reader to the current situation regarding interoperability matters, more specifically on standards and data models that are contemplated in the project and are evaluated against ontologies exploring different tools to accomplish this goal.

Chapter 3 provides information regarding the components of the Platform Open Framework and the improvements in terms of communication.

Chapters 4 present how best practices and also security aspects are addressed in the final implementation of the platform components.

And finally, Chapter 5 discusses the conclusions of the deliverable.

1.4 How to Read this Document

This deliverable contains the second version of the report regarding the intra-platform communication between the three core components of the Platone Open Framework (Blockchain Access Layer, DSO Technical Platform, and Market Platform) and further external systems. A first version of this report can be found in D2.9[16].

The updates are presented in the document with a strict focus on Platone Platforms implemented within WP2. Other external systems, implemented within other WPs (WP3, WP4 and WP5) are briefly described, too. A greater level of detail is included in deliverables D3.9 [33], D4.5 [34] and D5.7 [35]. Use cases and scenarios, used for defining the functional and non-functional requirements, are available in D1.1 [3], D4.1 [4], and D5.2 [5]. Final reports regarding the implementation and integration of Platone can be found in D4.5 and D5.7. A more coherent overview of the Platone Open Framework is provided in the deliverable on Platform requirements and reference architecture, D2.1 [2], in the deliverables that contain the final versions of the reference implementations of the platform components in the field trials, D2.4 [6], D2.8 [7], D2.13 [8] and in the deliverable regarding the last version of the Framework integration D2.16 which will be delivered at the same time as this report.

2 Data Interoperability

This section provides an overview of different data formats that are used in the Platone field trials and needed to be integrated into the different Platforms of the Platone Open Framework. Subsection 2.12.1 presents relevant standardized or proprietary data formats the Platone Framework receives from devices or external systems. Subsection 2.22.2 presents the evaluation of some payloads contained in data models with relevant ontologies that are considered for semantic representation of data within the Platone framework.

2.1 Standards and data formats relevant for the field trials

This subsection presents data formats and related standards that are used in the second release and integration phase of the Platone Open Framework in the field trials.

Deliverable 2.9 already gave insights into the specific data models that are supported by the Platone Open Framework and some benefits of their usage. It is important to recall the mentioned data models which can be summarized in the following paragraphs.

The IEC 61968 and IEC 61970 series of standards are both based on the Common Information Model (CIM). The IEC 61968 distribution management standard is the first part in a series of standards that define interfaces for the major elements of an interface architecture for Distribution Management Systems. The data model standard CIM IEC 61968-9 [11] is used for measurement data of smart meters and serves to standardize the integration of metering systems with other interested parties. It is noteworthy that the standard does not define specific communication protocols over which these data are exchanged.

Figure 1 shows an example of IEC 61968-9 CIM meter readings data in XML format as used in Work Package (WP) 4 to transfer aggregated metering data sets from automated meter reading systems into the Blockchain Access Layer of the Platone Open Framework.


```xml
<mr:MeterReadings xmlns:mr="http://iec.ch/TC57/2011/MeterReadings#">
   <mr:MeterReading>
      <mr:Meter>
         <mr:Names>
            <mr:name>11111111</mr:name>
            <mr:NameType>
               <mr:name>EndpointID 110</mr:name>
               <mr:NameTypeAuthority>
                  <mr:name>NAME</mr:name>
               </mr:NameTypeAuthority>
            </mr:NameType>
         </mr:Names>
      </mr:Meter>
      <mr:Readings>
         <mr:timeStamp>2019-09-13T15:30:00+03:00</mr:timeStamp>
         <mr:value>11.04</mr:value>
         <mr:ReadingType ref="0.26.0.0.1.1.12.0.0.0.0.0.0.0.0.224.3.72.0"/>
      </mr:Readings>
      <mr:Readings>
         <mr:timeStamp>2019-09-13T15:30:00+03:00</mr:timeStamp>
         <mr:value>3.45</mr:value>
         <mr:ReadingType ref="0.26.0.0.1.1.12.0.0.0.0.0.0.0.0.224.3.73.0"/>
      </mr:Readings>
      <mr:UsagePoint>
         <mr:mRID>111111</mr:mRID>
      </mr:UsagePoint>
   </mr:MeterReading>
</mr:MeterReadings>
```


Figure 1: Meter Readings Data Model

In Platone use cases, a proprietary data format is used for data that is sent from a Low-Cost Phasor Measurement Unit (LOCO PMU) and is integrated into the Platone Blockchain Access Layer by means of MQTT. An example of a payload following this data model is shown in Figure 2.


```
{
    "device": "pmu1",
    "timestamp": "2020-05-20T10:27:57.980802+00:00",
    "readings": [
        {
            "component": "BUS1",
            "measurand": "voltmagnitude",
            "phase": "A",
            "data": 11
        },
        {
            "component": "BUS2",
            "measurand": "voltmagnitude",
            "phase": "A",
            "data": 22
        }
    ]
}
```


Figure 2: Example for PMU Reading Data Model

Regarding process standardization, it is worth noting that use cases were defined using a template based on the Use Case Methodology outlined in IEC 62559-2 [12]. Leveraging established methodologies such as the one presented in IEC 62559 leads to well-defined use cases with comprehensive elaboration of their interrelationships, resulting in a coherent demo narrative. The adoption of standardized processes not only ensures consistency and efficiency but also brings significant benefits for interoperability. By following this methodology, teams involved in the project can achieve a shared understanding of the use cases, facilitating collaboration and integration. This alignment with established methodologies enhances interoperability right from the process definition phase, promoting better interactions and exchange of information across diverse systems and stakeholders.

2.2 Ontologies

Semantic solutions can be achieved through the utilization of ontologies like the Smart Appliance REFerence (SAREF) [13], a reference ontology designed for smart appliances, serving as an umbrella to enhance the integration of semantic data across diverse domains within the Internet of Things (IoT). Moreover, SAREF can be further extended to incorporate additional concepts, as demonstrated by its extension SAREF4ENER which extends to energy-related concepts. In addition, the SmArt eneRGy dOmain oNtology (SARGON)[14] ontology incorporates more classes, properties, and instances, encompassing concepts from the building and electrical grid automation domain.

These ontologies play a crucial role in defining concepts that can facilitate semantic annotation, which is a process of enriching data payloads with relevant information. The Semantic Annotation Service to Integrate Smart Energy Data (SISEG)[15] is a tool that accepts inputs in form of data (JSON or XML formats) and also an ontology, and is capable of automatically annotating these payloads with tags aligned to the specific ontology. By utilizing SARGON ontology and a tool like SISEG, payloads can be semantically annotated to enhance interoperability. After this annotation, a score can be calculated to determine the number of valid annotated attributes. This process is considered semi-automatic due to the ongoing evolution and research of these technologies necessitating mechanisms to identify and measure potential errors. While human intervention remains essential to verify data annotations, this valuable input can be used to improve the model and enhance accuracy during this phase of development, which in the end will benefit the Platform by having more reliability in giving a context to the payloads.

SARGON was intended to be evaluated on a subset of data models and different approaches can be considered to reach this goal. As an example of using the above-mentioned SISEG tool, two payload are selected representing respectively the meter data shown in Figure 1 and the PMU data (which is using a proprietary data model) shown in Figure 2. Then an alignment is tested utilizing the mentioned annotation tool SISEG and with the ontologies SARGON and SAREF. This automation tool, capable of classifying and matching terms, has a very valuable function to evaluate ontologies with data models or payloads. The results can be seen in Figure 3 and Figure 4 respectively for each payload mentioned before.


```
{
    "@context": {
    "MeterReading": "https://w3id.org/saref#EnergyMeter",
    "meter": "https://w3id.org/saref#Meter",
    "name": "https://sargon-n5geh.netlify.app/ontology/1.0/object_properties/name",
    "readings": "https://w3id.org/saref#EnergyMeter",
    "usagepoint": "https://w3id.org/saref#Profile"
    },
    "@id": "Meter",
    "@type": "https://w3id.org/saref#Meter",
    "MeterReading": {
    "meter": {
    "names": {"name": {
    "@type": "https://sargon-n5geh.netlify.app/ontology/1.0/object_properties/name",
    "@value": "11111111"
    },
    "nametype": {"name": {
    "@type": "https://sargon-n5geh.netlify.app/ontology/1.0/object_properties/name",
    "@value": "Endpoint ID 110"
    },
    "nametypeauthority": {"name": {
    "@type": "https://sargon-n5geh.netlify.app/ontology/1.0/object_properties/name",
    "@value": "NAME"
    }}}},
    "@type": "https://w3id.org/saref#Meter"
    },
    "readings": {
    "@type": "https://w3id.org/saref#EnergyMeter",
    "@value": [
    {
    "timestamp": "2019-09-13T15:30:00+03:00",
    "value": "11.04",
    "ReadingType": {
    "@ref": "0.26.0.0.1.1.12.0.0.0.0.0.0.0.224.3.72.0"
    }},
    {
    "timestamp": "2019-09-13T15:30:00+03:00",
    "value": "3.45",
    "ReadingType": {
    "@ref": "0.26.0.0.1.1.12.0.0.0.0.0.0.0.224.3.73.0"
    }}]},
    "usagepoint": {
    "@id": "111111",
    "@type": "https://w3id.org/saref#Profile"
    },
    "@type": "https://w3id.org/saref#EnergyMeter"
    }
}
```


Figure 3: Meter reading payload annotated with SISEG and ontologies

```json
{
"@context": {
"device": "https://w3id.org/saref#Device",
"timestamp": "https://sargon-n5geh.netlify.app/ontology/1.0/classes/Timestamp",
"readings": "https://w3id.org/saref#MeteringFunction"
},
"@id": "PMU1",
"@type": "https://sargon-n5geh.netlify.app/ontology/1.0/classes/PMU",
"device": {
"@id": "device",
"@value": "PMU1",
"@type": "https://w3id.org/saref#Device"
},
"timestamp": {
"@id": "timestamp",
"@value": "2020-05-20T10:27:57.980802+00:00",
"@type": "https://sargon-n5geh.netlify.app/ontology/1.0/classes/Timestamp"
},
"readings": {
"@id": "readings",
"@value": [
{
"component": "BUS1",
"measurand": "voltmagnitude",
"phase": "A",
"data": 11
},
{
"component": "BUS2",
"measurand": "voltmagnitude",
"phase": "A",
"data": 22
}
],
"@type": "https://w3id.org/saref#EnergyMeter"
}
}
```


Figure 4: PMU Reading Data Model annotated with SISEG and ontologies

Figure 3Figure 3 and Figure 4Figure 4, show data in JSON-Linked Data format, as this format can play a crucial role in enhancing interoperability by presenting structured data with embedded semantic annotations. This representation allows different systems and applications to understand and exchange data in a standardized format, making it easier to interpret and process the information. By using Resource Description Framework (RDF) triples and Uniform Resource Identifiers (URIs), JSON-LD links data to existing ontologies and knowledge graphs. This linking creates a shared understanding of the data's meaning by providing a context to the payloads, and with this facilitating seamless data exchange between heterogeneous systems. It can be seen in the last two figures presented above that some concepts like "Timestamp" and "PMU" are pointing to the specific concepts inside the SARGON pool of resources, while other concepts like "Device", "Meter", and "EnergyMeter" are pointing to the specific concepts inside the SAREF pool of resources.

The automatic annotation process offers valuable assistance in verifying the context of metadata of payloads, which can subsequently be utilized to provide an accurate vocabulary database that groups together similar, pre-existing data based on established models. Additionally, leveraging semantic similarity techniques enables effective schema matching, facilitating the identification of meaningful connections between different schemas or data structures. The DSO Technical Platform design builds on previous work done in the Horizon 2020 project SOGNO [30]. A service that provides semantic similarity was added to the SOGNO platform and can bring benefits to the Platone project improving data integration and interoperability. By comparing data based on their semantic similarities, this service can identify relevant relationships and mappings between various data elements, promoting seamless data exchange and fostering a deeper understanding of the data. This enhances the platforms' capability to handle diverse data sources, creating a more robust and comprehensive system for data processing and analysis. Furthermore, the use of semantic similarity aids in maintaining data consistency and accuracy, reducing potential conflicts or errors in data interpretation, thus contributing to the overall reliability and efficiency of the platform.

3 Platone Open Framework Design and Communication Enhancements

Platone Open Framework is an open source and modular framework based on blockchain technology that enables a secure and shared data management system, allows standard and flexible integration of external solutions (e.g., legacy solutions), and is open to integration of external services through standardized open application program interfaces (APIs) [2].

The framework main characteristics rely heavily on the concepts of interoperability, adaptability, and flexibility and for these reasons it includes by design several interoperable and standardised mechanisms for the integration.

More in detail, the Framework consists of three main components, each of these can be used individually, or integrated with the others.

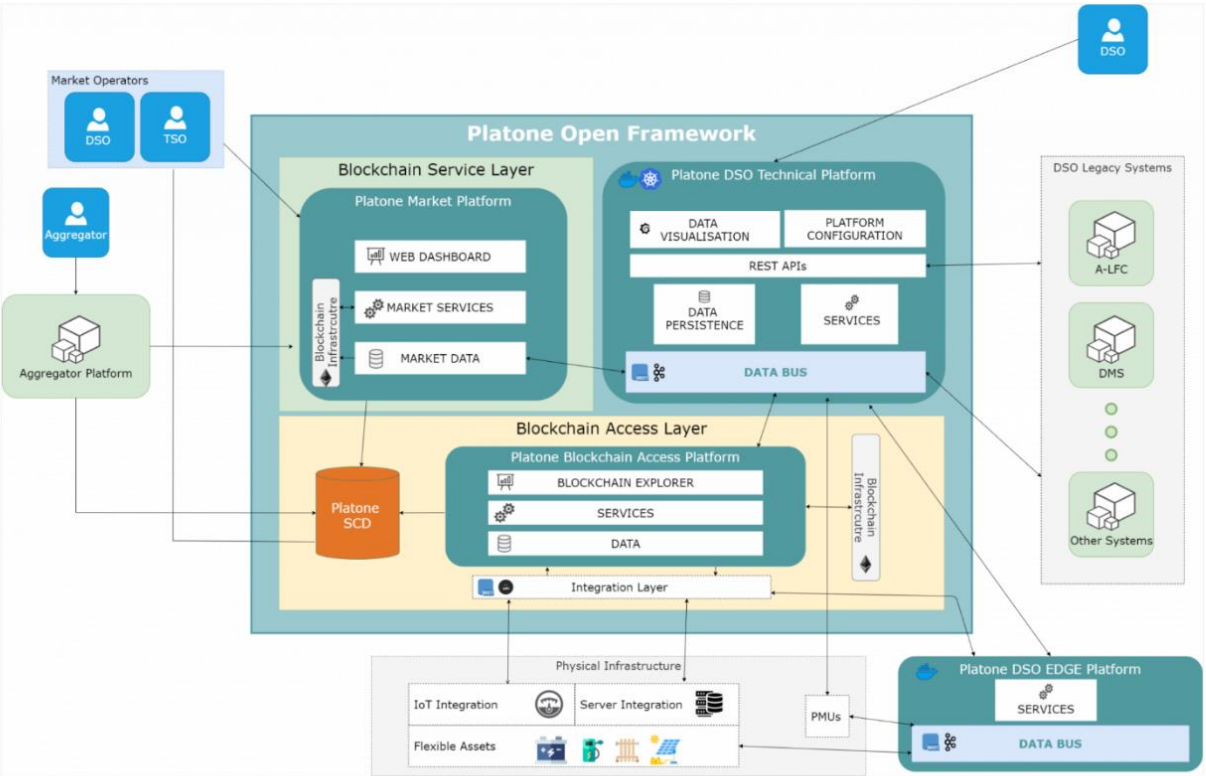

Figure 5: Platone Open Framework

Platone Market Platform, which allows the support of wide geographical area flexibility requests from TSOs and local flexibility requests from DSOs. These are matched with offers coming from aggregators, resolving conflicts according to pre-defined rules of dispatching priorities. All the market operations are registered and certified within the blockchain service layer, ensuring a transparency, security, and trustworthiness among all the market participants.

Blockchain Access Layer, which adds a further level of security and trustworthiness to the framework. It is an extension of the physical infrastructure and performs multiple tasks, among which are data certification and automated flexibility execution through Smart Contracts. It includes the Blockchain Access Platform and the Shared Customer Database (SCD).

Platone DSO Technical Platform, which allows DSOs to manage the distribution grid in a secure, efficient, and stable manner. It is based on an open-source extensible microservices platform and allows to deploy, as Docker containers, specific services for the DSOs and execute them on Kubernetes. The Data Bus layer, included on the DSO Technical Platform (DSOTP), allows the integration of other components of the Platone framework and of external components (e.g., DSO Management System) with a direct connection to the classical Supervisory Control and Data Acquisition (SCADA) system adopted by the DSO and served by standard communication protocols.

In the next subsections, a more detailed description of the interoperability and standardisation of the different components is provided.

3.1 Blockchain Access Layer

Platone Blockchain Access Layer (BAL) is the component in charge of enabling a standard, secure, and easy integration of energy data coming from the physical infrastructure and of granting access to this data to DSOs and other energy stakeholders.

As already described in D2.9, it implements a data integration interface for collecting several kind of energy data coming from smart meters (e.g., PMU and Light Node), external servers (e.g., DSO Data Server) supporting standardised data models and certifying this data thanks to blockchain technology and smart contracts, ensuring data integrity, and avoiding data tampering.

The energy data collected, harmonized, and certified are available for all the energy stakeholders involved as well as external platforms and services (e.g., the DSOTP) which need to use this data within the SCD.

The BAL supports MQTT as main protocol for the integration of the data coming from the IoT devices and external servers, as well as for sharing the collected to the external stakeholders (through the SCD interface). In addition, Hypertext Transfer Protocol Secure (HTTPs) and Representational State Transfer (REST) Application Programming Interface (APIs) are also supported by the SCD for retrieving data in a synchronous way.

As depicted in Figure 6, BAL supports and facilitates the entire data management process. The final version of the BAL, not described in the previous version of this deliverable, includes now an innovative mechanism for identity management and access control (Data Access Policies, DAP), based on blockchain technology. More details are provided in Ch. 4.2.

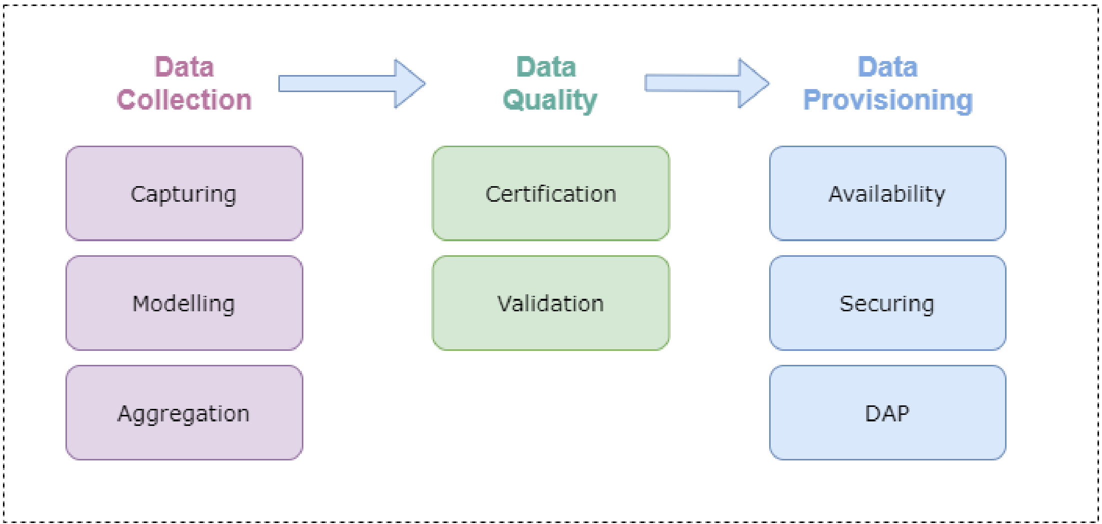

Figure 6: BAL Data Management Process

3.2 Market Platform

The Platone Market Platform is one of the core components of the Platone Open Framework. It is a blockchain-based platform that enables the management of the flexibility market, involving any possible stakeholders (TSOs, DSOs, Aggregators, Customers).

All the market operations are registered and certified within the blockchain service layer ensuring a higher level of transparency, security, and trustworthiness among all the market players.Furthermore, the Platone Market Platform enable an innovative incentivisation mechanism for customers engagement based on blockchain technology, smart contracts, and tokenisation.

All the aspects related to the interoperability and security, including REST APIs and Oauth 2.0 for synchronous communication and Apache Kafka with Two-Way authentication for asynchronous one, were described in D2.9 and no relevant updates were made in the final release of the Platone Market Platform under this point of view.

3.3 DSO Technical Platform

The Platone DSO Technical Platform enables distribution system operators to fulfill market requests by evaluating the current grid state and activating local flexibility requests while ensuring the reliability and operational quality of services by enlarged grid observability. The platform design builds on previous work done in the Horizon 2020 project SOGNO [30] and relies massively on a micro-service architecture in which a DSO can easily deploy additional services onto the platform. The first prototype of the DSOTP is described in Platone D2.6 and incremented updates of the platform can be seen in D2.7 for the second version and 2.8 for the third and last version.

To address requirements such as high availability, scalability, and modularity from the very beginning, the DSOTP is designed for deployment on Kubernetes [28] clusters. Kubernetes, also known as K8s, is an open-source system for automation deployment, scaling, and management of containerized applications. As all micro-services of the platform are per requirement containerized in Docker [29] containers, they can easily be deployed on a Kubernetes cluster. Kubernetes also simplifies different deployment approaches: from edge- and public-cloud to on-premises installation. However, the on-premises installation is considered the most relevant for a control centre platform. In order to minimize initial hurdles, Platone provides detailed installation manuals for a local installation based on the lightweight Kubernetes distribution k3s [38].

Figure 7 illustrates the architecture of the DSOTP. The data bus is one of its core components and is implemented by means of a message broker to which all services can publish and/or subscribe in order to exchange data with other services and field devices or with external systems. Field devices or external systems can be made available in the data bus either directly or through the Platone BAL [8].

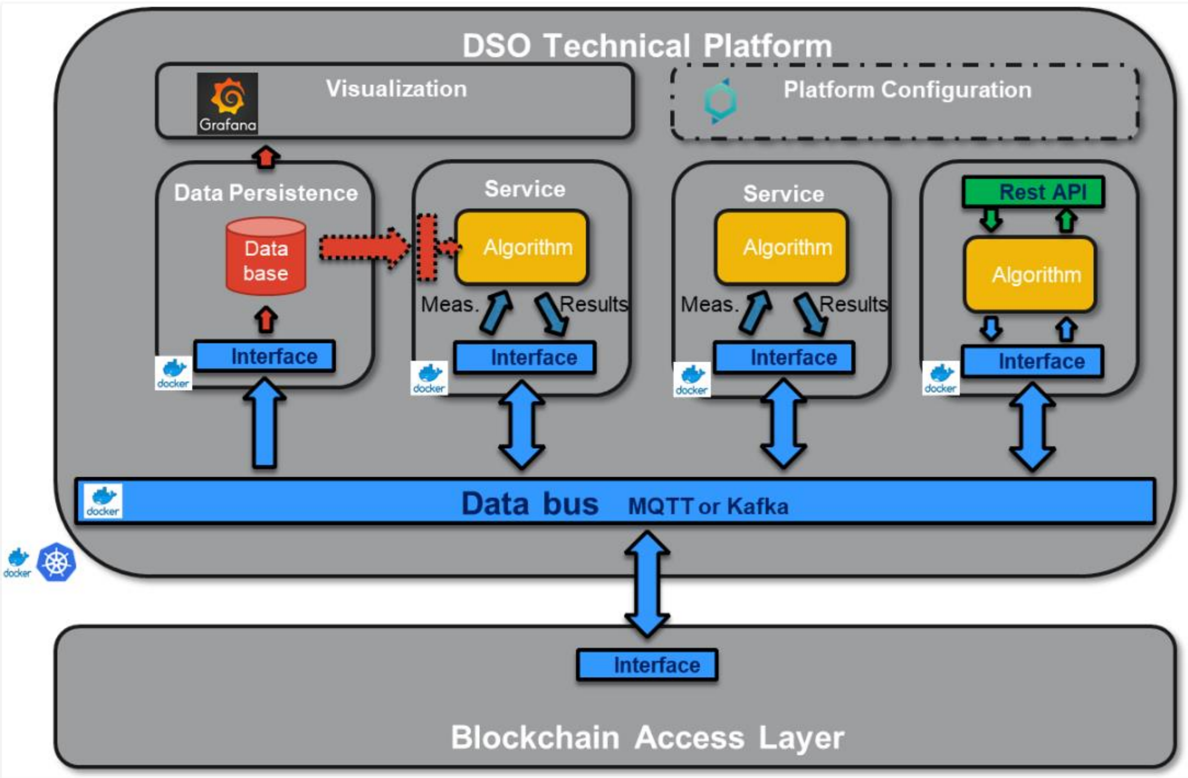

Figure 7: Platone DSO Technical Platform Architecture

During the last version of the DSOTP, the balancing service was redesigned and renamed as Python Microgrid Flexibility Management service (pymfm) to be more recognizable as a concrete software solution. This service exposes HTTP endpoints on an API, which is completely documented using the OpenAPI 3.0 standard, which contributes to improved communication, collaboration, and interoperability by allowing to discover and understand easily the capabilities of the service. The pymfm service is executed when data is sent to well-defined URLs that are designed to be used intuitively. In this regard, the endpoints to create and list jobs are executed by sending requests to "<hostname>/balancing/", and to handle specific jobs, the request is made to the endpoint "<hostname>/balancing/<ID>".

In general and to ensure an easy and efficient deployment, a Dockerfile is provided for each new service of the open source version of DSOTP, i.e., SOGNO. Furthermore, to foster continuous integration and version control, a configuration for GitlabCI is thoughtfully provided. This configuration automatically rebuilds the Docker image whenever any changes are made to the repository's main branch, thus preventing any version discrepancies and promoting a unified codebase. The resulting image is then securely pushed to the project's Gitlab registry, ensuring easy accessibility for authorized users. This process empowers developers to quickly deploy and access the latest version of the service, fostering a collaborative and agile development environment.

For deployment to a Kubernetes cluster, resource definitions are provided for each service. These resource definitions are essential components that outline the specifications and configuration details for each service. By providing these resource definitions, it is ensured that each service can be easily deployed, managed, and scaled within the Kubernetes cluster.

3.4 External Systems

Regarding the integration of the core platform components of the Platone Open Framework (i.e. D2.15[9]) the Framework allows the integration of external systems. This section incorporates insights of the ALF-C and the Greek demo integrations from the WP4 and WP5 field trials.

3.4.1 Avacon Local Flex Controller (ALF-C)

The ALF-C is the main component developed, implemented, and tested in the German field trial of WP5 whose implementation is based on Microsoft Azure Functions and Function Apps. It integrates some of the assets of the field trial such as the large-scaled battery energy storage system and domestic storages (owned by households in combination with a rooftop photovoltaic system). In addition, it is combined with the Platone Open Framework, to integrate the PMU at the substation level via the BAL, and to utilize an external optimization algorithm that is provided as a service by the DSOTP.

In the latest update of DSOTP, improvements have been made to the REST API resulting in enhanced integration capabilities. Specifically, to streamline its core business logic, ALF-C utilizes REST API calls to access an optimization algorithm within pymfm service that operates independently within the DSOTP. Furthermore, the data visualization has been shifted to the Azure platform of the German Demo. As a result, the PMU data originating from PMU devices are integrated and certified into the Platone BAL, and it is now directly visualized on the ALF-C, eliminating the previous reliance on the DSOTP for data visualization purposes.

The exchange of data between the German Demo and the DSOTP is done by using distinct approaches. DSOTP leverages the MQTT protocol with Secure Sockets Layer encryption to transmit data that is previously pre-processed, to the German Demo Azure endpoint, ensuring real-time telemetry and responsiveness. Meanwhile, the German demo securely communicates with the pymfm service via HTTPS, utilizing JSON payloads for structured exchanges. The combination of MQTT's efficiency and stream-like connections for real-time data and HTTPS's secure request-response interactions enables seamless data exchange, allowing for optimized operations and enhanced system efficiency.

In this particular data exchange scenario, the data does not necessarily reside in the DSOTP. As a result, the decision to store the data within the specific platform of the external system (in this case the own Azure platform of the German demo) enhances the security measures significantly. By retaining control of the data within de German Demo server, it becomes easier to implement robust security protocols such as access controls, encryption, and auditing mechanisms designed to safeguard sensitive information. This approach minimizes the potential attack surface and reduces the risk of data exposure during transit. By prioritizing the secure storage of data within the platform of external systems, a strong foundation for reliable and protected data interchange is established.

3.4.2 Greek Demo and the State Estimation Tool

As part of the Platone project, PMU devices were installed at the Greek demonstration site. In its most recent development, phasor measurement service is provided by the so-called Synchronized Measurement Unit (SMU). SMU functions as a LOCO PMU installed at the demonstration site with open-source hardware design and software code and can provide different synchronized measurements.

In this scenario, SMUs are configured to monitor channels with three-phase voltages and three-phase currents. A dedicated service is utilized to handle the resulting phasors, and it generates measurements in XML data format that adhere to CIM (Common Information Model) standards. Within this process, the measurements are integrated with further DSOTP layers, including the state estimation service.

The State Estimation (SE) tool is able to run receiving joined information from the SCADA, AMR, and PMUs via the databus. In order to incorporate the SE tool into the Greek Demo, it was necessary to create measurement and topology converters. The Greek demo's electricity distribution network topologies and real-time measurements are stored in XML format, following the Common Interface Model (CIM) standard. However, the State Estimation (SE) tool requires input files in a different format (PSS/E format). To make the data compatible with the SE tool, intermediate converters were developed using Python and Matlab. Figure 8 shows the flow diagram of the converters. These converters adjust the data to meet the specific requirements of the SE tool, enabling it to read and process the information in a proper format. For further explanation in this regard, the reader can refer to D4.5 [34].

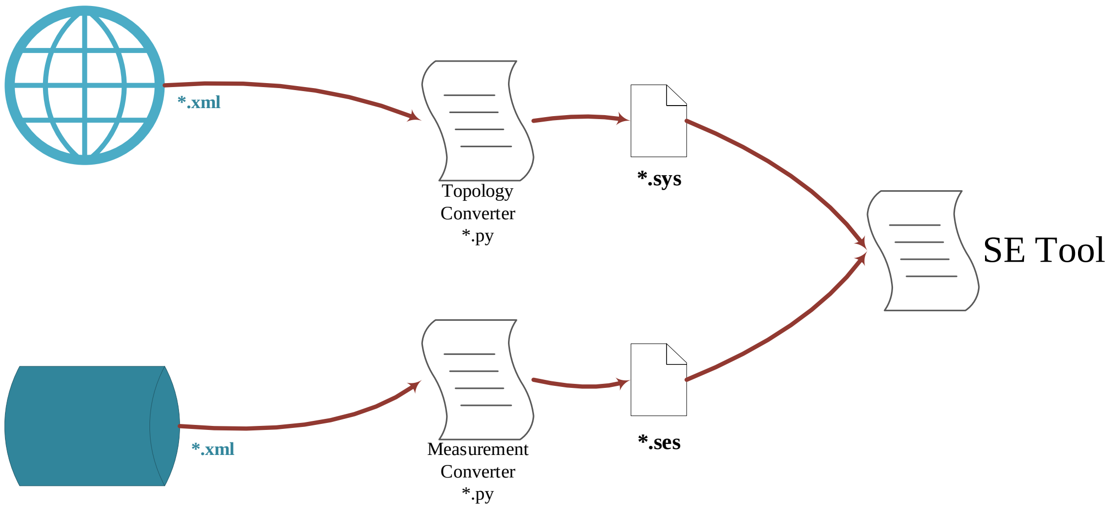

Figure 8: Flow diagram of the conversion.

In a later stage, the transition of the SE tool code from Matlab to Octave was necessary in order to tackle the problem of Matlab licensing. In addition, a graph representation of the network is generated in a dashboard utilizing the information from the CIM data, which is the data format provided by the Greek demo. A static graph layout was also created to parse the XML files and to enable intuitive access to the results of state estimation by giving an overview of the network as a whole.

4 Secure Intra-Platform Communication

This section provides an overview of security design principles and best practices that were considered in the final release cycle and integration phase of the Platone open framework.

Additional security features that were foreseen for this release are described in this section and these include a more unified approach to Identity Management (IDM) and Access control between the different Platforms.

4.1 Design Principles

For communication within the Platone Open Framework, synchronous and asynchronous communications interfaces exist. Synchronous communication is used for instance for configuring services running on one of the platforms or for querying specific data sets and is implemented by means of REST APIs. Asynchronous communication is used for instance for event-based or streaming-data mechanisms and is implemented by means of a message broker. Since the first release of the Framework, MQTT and Apache Kafka are used as internal message brokers in the different Platone platforms. Table 1 depicts internal and external communication protocols.

In this regard, the databus utilized in the DSOTP is considered a very central component and therefore switching entirely to Kafka requires a strategy. The approach, in this case, is to do a full implementation of native Kafka with new services that will be incorporated into the DSOTP. And for already used services, a trial is suggested to integrate them with a minimal or no impact on the interface coding of the established service. This strategy will work well and smoothly, as MQTT in this case will continue to be used as the internal broker during the transition. The starting development of Kafka for DSOTP which can be extended to be used for new services can be found in the repository of the project [39].

Table 1: Communication Protocol Matrix

Platform	Internal	External	Authentication Mechanism
Blockchain Access Layer	MQTT Kafka	MQTTS Kafka Transport Layer Security (TLS)	Basic Authentication
DSO Technical Platform	MQTT TLS HTTP	MQTTS HTTPS	OAuth2, Basic Authentication
Platone Market Platform	Kafka HTTP	Kafka TLS HTTPS	Oauth2, Client-Cert. Authentication

4.2 Identity Management and Access Control

The BAL is in charge to collect, harmonise and certify data in a secure way (through the integration layer) and at the same time to make this data available (through the Communication Layer) including specific mechanisms for access control and the secure provision of the data to the external actors and platforms.

For this reason, as described in D2.12 [31], the BAL includes a specific tool for the identity management and data access management and control, named Data Management Tool. This specific tool ensures to identify uniquely all the data provisioners and consumers as well as to all the energy stakeholders to handle the data provisioning and consumption in a secure and trusted way, defining specific DAP.

The first version of Data Management Tool, implemented in the second version of the BAL, was based on a low layer access control, exploiting the Mosquitto Dynamic Security Plugin [36] and the Mosquitto Access Control List [37] for defining and check access control rules.

Within the BAL, the blockchain technology play a key role and can address three important challenges:

- ensure data privacy,
- manage data access and usage control,
- incentivize secure data sharing.

For this reason, the final version of the BAL included an identity and data access management based on blockchain technology.

4.2.1 Blockchain-based identity and data access management

In the final version of the BAL, described in D2.13 [8], the Data Management Tool was extended in order to support the blockchain technology and Smart Contracts for implementing data sharing without losing control and ownership of it.

Through smart contracts, it is possible to track who shared what, with whom, when, by what means and for what purposes in a verifiable fashion.

In the final version of the Data Management Tool, all the topic subscriptions are converted to a specific format and registered within the blockchain technology using a specific smart contract. The smart contract registers the subscription declaration and allows to verify (at any moment) the status and the date of the subscription.

Figure 9 represents the new schema of subscription management, using blockchain technology.

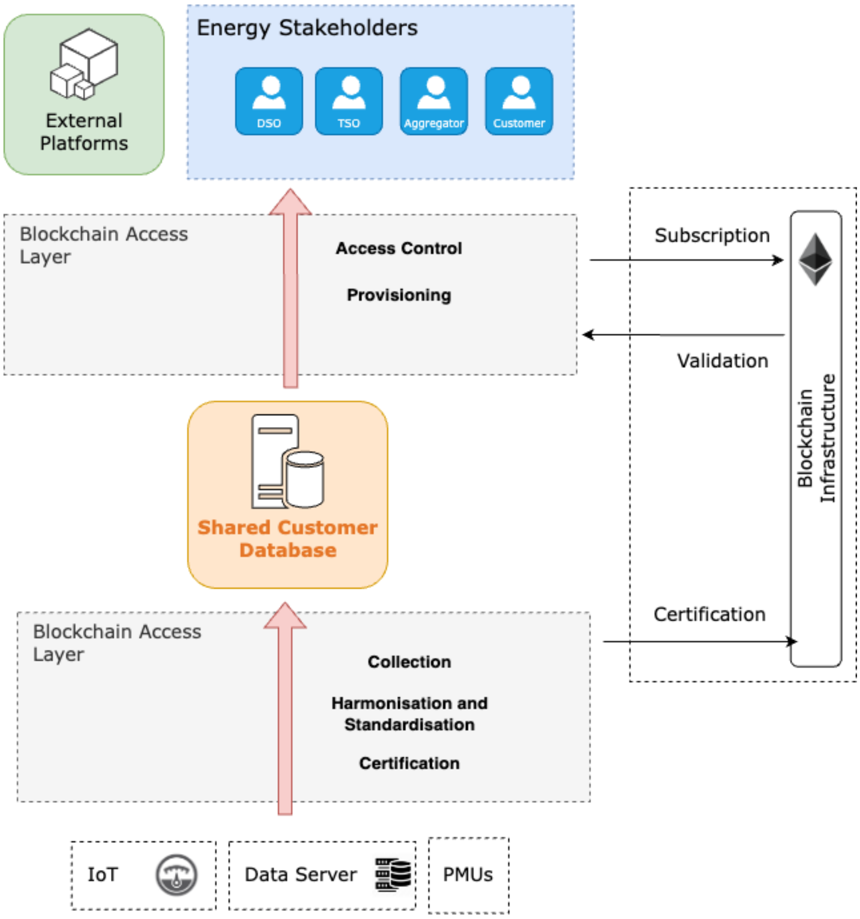

Figure 9: Data Management Tool (v2) with Blockchain and Smart Contracts

Smart Contract

The implemented smart contract contains the subscription declarations of the data actors (provider and consumer). It consists of the following properties:

- Provider Identity: wallet Address of the data provider
- Consumer Identity: wallet Address of the data consumer
- Topic: unique identifier for subscription topic
- Timestamp: timestamping proof of the subscription declaration
- Status: the status of the subscription declaration (accepted or rejected)

4.3 Best practices from Field Trials

Some remarkable best practices can be appreciated in the usage of technologies and will be described according to the final updates.

In the German Demo, Transport Layer Security (TLS) [24] is used to secure data streaming to their Azure platform. In this scenario, a front end is involved as a connection to the DSOTP, and security measures are reinforced by encrypted communication guaranteeing higher data integrity and privacy.

For the pymfm, in terms of communication, then developed API had been subject to an improvement to follow the REST standard and is automatically documented in the OpenAPI standard, specifically in its version 3.0.2. The endpoints were adjusted to be more robust, and validation was added for payloads.

In the Greek Demo, the standard CIM is used for the topology of the network and for the real-time measurements data to be transferred, from which a more extensive explanation can be found in D4.5. The use of standards provides consistency in data formats and structures promoting clear communication and accurate interpretation between the different components. This makes it easy to perform input validation, data integrity checks and helps in terms of access control by providing a uniform interpretation of data to determine permission access to resources.

During the SMUs installation phase, challenging tasks of current and voltage signals conversion and various integration aspects were conducted. To facilitate code maintenance and provision of additional services, the SMUs are connected to a virtual network (Figure 10). This connection allows for seamless collaboration between SMUs and other systems and provides access to necessary resources to support the commissioning and installation process. The DSOTP is installed and isolated from the external environment.

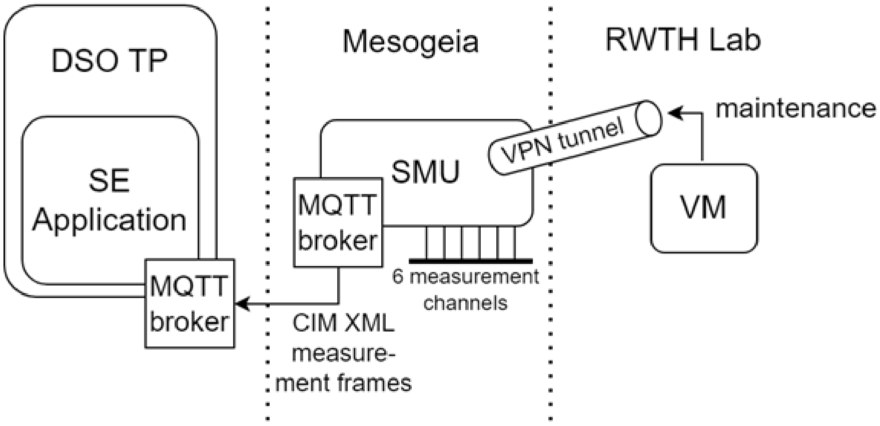

Figure 10: Connection of SMU in Mesogeia setup.

4.4 Outlook for future work

In the future, it is expected that the repositories of the DSOTP and the open-source SOGNO to be analysed for their unification into one main repository. During this unification, an organization of the DSOTP repositories is foreseen, at least for the centralization of common features, like authentication or user management.

As security is considered a very important aspect to take into account within the Platform, and as there are a variety of different decoupled services that need to be maintained in the suit of the platform, a security team or task force is recommended to help on improving the security by double checking and providing expertise to developers and implementers.

5 Conclusion

The work done at this stage, allowed us to present the final updates regarding the intra-platform communication of the Platone Open Framework from different points of view, in order to allow all stakeholders involved to understand the characteristics and consider possible integrations.

This report gave an overview of the data models that were used in the field trial of the Platone Open Framework. The report showed payloads conforming to the IEC 61968-9 CIM meter readings data model and a proprietary data format used for LOCO PMU, that were evaluated with the SARGON ontology. For this evaluation, the SISEG tool was used to annotate the payloads automatically. The results showed that there are some concepts like "Timestamp" and "PMU", "Device", "Meter", and "EnergyMeter that are aligned with the SARGON and SAREF ontologies and could be used to annotate payloads to enhance interoperability. In addition, a semantic similarity tool that was integrated into SOGNO as a service was also presented as a possible solution to enhance interoperability.

Furthermore, the document described the relevant updates in terms of communication and design regarding the integration with services provided by the DSOTP. In this regard, with concern to the German demo, it described improvements to the APIs and a shift in the data visualization from the DSOTP to Azure platform, and concerning the Greek demo, it described the development of converters needed to integrate the state estimation tool and a dashboard to represent the graph of network.

The communication channels for the intra-platform communication are protected following well-established industry standards for cyber security. Moreover, the final version provided an extension of the BAL's Data Management Tool and now includes an innovative mechanism for identity management and access control, based on blockchain technology and Smart Contracts for implementing data sharing without losing control and ownership of it.

Finally yet importantly, all updates of the communication protocols are also included in the public open-source git repositories of the respective Platone Open Framework components [39].

6 List of Tables

7 List of Figures

8 List of References

[1] European Commission, “2050 long-term strategy”, [Online]. Available: https://ec.europa.eu/clima/policies/strategies/2050_en

[2] Platone “D2.1 Platone Platform requirements and reference architecture (v1)” [Online]. Available: https://platone-h2020.eu/

[3] Platone “D1.1 v1.0 General functional requirements and specifications of joint activities in the demonstrators” [Online]. Available: https://platoneh2020.eu/

[4] Platone “D4.1 Report on the definitions of KPIs and UCs” [Online]. Available: https://www.platone-h2020.eu/

[5] Platone “D5.2 Detailed use case descriptions” [Online]. Available: https://www.platone-h2020.eu/

[6] Platone “D2.4 PlatOne Market platform (v2)” [Online]. Available: https://www.platone-h2020.eu/

[7] Platone “D2.8 PlatOne DSO Technical Platform (v3)” [Online]. Available: https://www.platone-h2020.eu/

[8] Platone “D2.13 PlatOne BlockChain Customer Access Layer (v3)” [Online]. Available: https://www.platone-h2020.eu/

[9] Platone “D2.15 PlatOne Integrated Framework Prototype (v1)” [Online]. Available: https://www.platone-h2020.eu/

[10] Uslar, M., Specht, M., Rohjans, S., Trefke, J., Gonzalez, J.M.V., The Common Information Model CIM: IEC 61968/61970 and 62325 –A practical introduction to the CIM (Power Systems), Springer, 2012.

[11] IEC 61968-9 : “Application integration at electric utilities - System interfaces for distribution management - Part 9: Interfaces for meter reading and control”, 2013

[12] IEC 62559-2: “Use case methodology”. [Online]. Available: https://webstore.iec.ch/publication/22349

[13] “SAREF ontology” [Online]. Available: http://ontology.tno.nl/saref

[14] Haghgoo, Maliheh, et al. "SARGON–Smart energy domain ontology." *IET Smart Cities* 2.4 (2020): 191-198.

[15] SiSEG-Auto Semantic Annotation Service to Integrate Smart Energy Data. [Online]. Available: http://publications.rwth-aachen.de/record/841589/files/841589.pdf

[16] Platone “D2.9 Specification of the interoperability and standard communication protocols (v1)”. [Online]. Available: https://www.platone-h2020.eu/

[17] “Web Ontology Language (OWL)” [Online]. Available: https://www.w3.org/OWL/

[18] W3C “RDF 1.1 Turtle” [Online]. Available: https://www.w3.org/TR/turtle

[19] Apache “Apache Cassandra | Apache Cassandra Documentation” [Online]. Available: https://cassandra.apache.org/_/index.html

[20] OASIS “MQTT - The Standard for IoT Messaging” [Online]. Available: https://mqtt.org/

[21] Eclipse Foundation “Eclipse Mosquitto” [Online]. Available: https://mosquitto.org/

[22] OpenJS Foundation “Express - Node.js web application framework” [Online]. Available: https://expressjs.com/

[23] Apache Software Foundation “Apache Kafka” [Online]. Available: https://kafka.apache.org/

[24] Internet Society “What is TLS & How Does it Work? | ISOC Internet Society” [Online]. Available: https://www.internetsociety.org/deploy360/tls/basics/

[25] ethereum.org “Introduction to smart contracts | ethereum.org ” [Online]. Available: https://ethereum.org/en/developers/docs/smart-contracts/

[26] Confluent “Configure Confluent Platform to use TLS/SSL encryption and authentication | Confluent Documentation” [Online]. Available: https://docs.confluent.io/platform/current/kafka/authentication_ssl.html

[27] Linux Foundation Energy “SOGNO - LF Energy” [Online]. Available: https://www.lfenergy.org/projects/sogno/

[28] “Kubernetes” [Online]. Available: https://kubernetes.io/

[29] Docker Inc., “ Empowering App Development for Developers | Docker ” [Online]. Available: https://www.docker.com/

[30] “K3s: Lightweight Kubernetes” [Online]. Available: https://k3s.io/

[31] PlatOne BlockChain Customer Access Layer (v2). [Online]. Available: https://www.platone-h2020.eu/

[32] Platone “D4.2 State estimation tool” [Online]. Available: https://www.platone-h2020.eu/

[33] Platone “D3.9 Report on main results achieved in the field test” [Online]. Available: https://platone-h2020.eu/

[34] Platone “D4.5 Mesogeia demonstration: report” [Online]. Available: https://platone-h2020.eu/

[35] Platone “D5.7 Final report” [Online]. Available: https://platone-h2020.eu/

[36] «Mosquitto - Dynamic Security Plugin,» [Online]. Available: https://mosquitto.org/documentation/dynamic-security/.

[37] «MQTT authentication & authorization, » [Online]. Available: https://www.chirpstack.io/project/guides/mqtt-authentication/.

[38] “Platone - Educational Video Series” [Online]. Available: https://www.platone-h2020.eu/Resources/Educational_Videos

[39] “Platone - Code Repository” [Online]. Available: https://git.rwth-aachen.de/acs/public/deliverables/platone

9 List of Abbreviations

Abbreviation	Term
ALF-C	Avacon Local Flexibility Controller
API	Application Programming Interface
BAL	Blockchain Access Layer
BAP	Blockchain Access Platform
CIM	Common Information Model
DB	Database
DSO	Distribution System Operator
DSOTP	DSO Technical Platform
EMS	Energy Management System
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICT	Information and Communication Technology
IDM	Identity Management
IEC	International Electrotechnical Commission
IoT	Internet of Things
JSON	JavaScript Object Notation
LOCO PMU	Low-Cost Phasor Measurement Unit
MQTT	Message Queuing Telemetry Transport
PMU	Phasor Measurement Unit
REST	Representational State Transfer
SAREF	Smart Appliance REFerence
SARGON	SmArt eneRGy dOmain oNtology
SCADA	Supervisory Control and Data Acquisition
SISEG	Semantic Annotation Service to Integrate Smart Energy Data
SMU	Synchronized Measurement Unit
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TSO	Transmission System Operator
WP	Work Package
XML	Extensible Markup Language