



OneNet Reference Platform First Release

D6.5

Authors:

Apostolos Kapetanios (ED)

Konstantinos Kotsalos (ED)

Ferdinando Bosco (ENG)

Helias Karagozidis (ED)

Sonja Happ (RWTH)

Maliheh Haghgoo (RWTH)

Anastasis Tzoumpas (UBE)

Kalle Kukk (ELE)

Vassilis Sakas (ED)

Distribution Level	Public
Responsible Partner	EUROPEAN DYNAMICS
Checked by WP leader	Vassilis Sakas (ED) - Date: ../07/2022
Verified by the appointed Reviewers	Eliska Trmalova (CEPS) - Date: 11/08/2022 Arvīds Jakušenoks (ST) - Date 01/08/2022
Approved by Project Coordinator	Stephan Gross (FhG) – Date: 15/08/2022

Dissemination Level		
PU	Public	X
CO	Confidential, only for members of the consortium (including the Commission Services)	
CI	Classified, as referred to in Commission Decision 2001/844/EC	



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957739



Issue Record

Planned delivery date	31/3/2022
Actual date of delivery	15/7/2022
Status and version	V0.1 – Final Version

Version	Date	Author(s)	Notes
0.1	11.07.2022	ED, ELER, ENG, RWTH, UBE	First integrated version of D6.5
0.2	18.07.2022	AK/ED	Added “7.3 Communication Channels”
		FB/ENG	Added Connector Description in chapter 7
0.3	21.07.2022	ED, ENG	Chapter re-arrangement
0.4	25.07.2022	FB/ENG	Added 7.2.3
		HK/ED	Added Chapter 9
		KK/ED	Added Annex (GUI Manual)
0.6	27.07.2022	VS/ED	Editing corrections
0.7	30.07.2022	ED	Corrections Chapter 9 and Annex
1.0	01.08.2022	ED	Final Cand. Version
1.1.	11.08.2022	ED	Final Version after reviews



About OneNet

OneNet will provide a seamless integration of all the actors in the electricity network across Europe to create the conditions for a synergistic operation that optimizes the overall energy system while creating an open and fair market structure.

The project OneNet (One Network for Europe) is funded through the EU's eighth Framework Programme Horizon 2020. It is titled "TSO – DSO Consumer: Large-scale demonstrations of innovative grid services through demand response, storage and small-scale (RES) generation" and responds to the call "Building a low-carbon, climate resilient future (LC)".

While the electrical grid is moving from being a fully centralized to a highly decentralized system, grid operators have to adapt to this changing environment and adjust their current business model to accommodate faster reactions and adaptive flexibility. This is an unprecedented challenge requiring an unprecedented solution. For this reason, the two major associations of grid operators in Europe, ENTSO-E and EDSO, have activated their members to put together a unique consortium.

OneNet will see the participation of a consortium of over 70 partners. Key partners in the consortium include: already mentioned ENTSO-E and EDSO, Elering, E-REDES, RWTH Aachen University, University of Comillas, VITO, European Dynamics, Ubitech, Engineering, and the EU's Florence School of Regulation (Energy).

The key elements of the project are:

1. Definition of a common market design for Europe: this means standardized products and key parameters for grid services which aim at the coordination of all actors, from grid operators to customers;
2. Definition of a Common IT Architecture and Common IT Interfaces: this means not trying to create a single IT platform for all the products but enabling an open architecture of interactions among several platforms so that anybody can join any market across Europe; and
3. Large-scale demonstrators to implement and showcase the scalable solutions developed throughout the project. These demonstrators are organized in four clusters coming to include countries in every region of Europe and testing innovative use cases never validated before.



Table of Contents

1 Introduction	9
1.1 OneNet scope	9
1.2 Report Outline	10
2 Aligning OneNet Development with other Tasks & Demos	11
2.1 Functional Description of OneNet Components	11
2.1.1 Task 6.1 Decentralized edge-level middleware for scalable platform agnostic data management and exchange	11
2.1.2 Task 6.2 Cross stakeholder Governance for Energy Data Exchange	19
2.1.3 Task 6.3 Data and Service Interoperability, Integration with FIWARE, and Homogenization Management	20
2.1.4 Task 6.4 AI, Big Data, IoT Workbench to support scalable near real time big data-enabled services	20
2.1.5 Task 6.6 Legal, Regulatory, Privacy and Cybersecurity Management & Compliance Tools	21
2.1.6 Task 6.7 Tools evolution management and Compliance to Reference Architecture	23
2.2 SUCs / Services to be tested in each Demo	23
3 Overview of the Integration Plan	31
4 Detailed Planning towards Integration	34
4.1 Phase A: First Development Phase (Decentralized edge-level middleware for scalable platform agnostic data management and exchange early prototype)	34
4.1.1 General Functional Requirements Definition - First version	34
4.1.2 Middleware development - First version	34
4.1.3 Integration & testing 1st cycle	35
4.2 Phase B: Second Development Phase (OneNet Reference Platform, Intermediate version)	35
4.2.1 General Functional Requirements Definition - Final version	35
4.2.2 Middleware development - Intermediate version	36
4.2.3 Workbench & Dashboard development	36
4.2.4 Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance first version	37
4.2.5 Ensure reference architecture compliance (first evaluation)	37
4.2.6 Integration & testing 2nd cycle	37
4.3 Phase C: Final Development Phase (OneNet Reference Platform, Final version)	37
4.3.1 Functional & Non-Functional Requirements finalization	38
4.3.2 Middleware development - Final version	38
4.3.3 Workbench & Dashboard development – Final version	38
4.3.4 Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance - Final version	38
4.3.5 Ensure reference architecture compliance (second evaluation)	39
4.3.6 Final Assessment	39
4.3.7 Integration & testing 3d cycle	39
5 Contingency Plan	41
6 DPIA Survey Methodology and Plan	45
6.1 Implementing DPIAs risk mitigation	45
6.1.1 User Registration and Consents	45
6.1.2 Technical measures	46
6.1.3 Organizational Measures	46
6.2 DPIA sample for OneNet	47



7 The OneNet Connector – Technical Characteristics	49
7.1 OneNet Connector Architecture	49
7.2 OneNet Connector – v1.0	50
7.2.1 OneNet GUI	50
7.2.2 OneNet Connector Data App	50
7.2.3 FIWARE and IDS based Components.....	53
7.2.4 REST-APIs	54
7.3 Communication Channels.....	55
8 Integration testing activities for OneNet	57
8.1 Introduction.....	57
8.2 Test Plan and Example Test Case Template	60
8.3 Frontend Testing	64
8.4 Backend Testing.....	64
8.5 Integration (Frontend-Backend) Testing	65
9 Deploying the OneNet Connector	67
9.1 H/W and S/W Prerequisites	67
9.2 Deployment	67
9.3 GUI Deployment	68
10 References	69
11 ANNEX – Graphical User Interface (GUI) Manual - v1.0	i
1 General overview.....	i
2 System administrator notes.....	i
3 OneNet users	ii
3.1 Accessing OneNet GUI	iii
3.2 Main screen	iv
3.3 Services.....	v
3.3.1 Cross-Platform Services	v
3.3.2 Requests.....	vii
3.3.3 My subscriptions	viii
3.4 Data exchanges.....	ix
3.4.2 Consume data	x
3.4.3 Completed data exchanges	xi
3.5 Connector Settings	xii
3.6 Documents & Downloads	xii



List of Abbreviations and Acronyms

Acronym	Meaning
AI	Artificial Intelligence
API	Application Programming interface
BO	Business Object
CDPSM	Common Distribution Power System Model
CGMES	Common Grid Model Exchange Standard
CIM	Common Information Model
CRUD	Create Read Update Delete
DB	Database
DEP	Data Exchange Platform
DoA	Description of Action
DPIA	Data Protection Impact Assessment
DSO	Distribution System Operator
ECCoSP	ENTSO-E's Connectivity and Communication Service Platform
ESMP	European Style Market Profile
FR	(Role) Flexibility Register
FSP	Flexibility Service Provider
FTC	FIWARE TRUE (TRUsted Engineering) Connector
GDPR	General Data Protection Regulation
GRTM	Governance Requirements Traceability Matrix
GUI	Graphical User Interface
HEMRM	Harmonized Electricity Market Role Model
HV	High Voltage
IDS	International Data Spaces
IDSA	International Data Space Association
IDSCP	IDS Communication Protocol
IoT	Internet of Things
LD	Linked Data
LV	Low Voltage
MO	Market Operator
MV	Medium Voltage
NFR	Non-functional Requirements
OCSP	Online Certificate Status Protocol
PnP	Plug-n-Play

RTM	Requirements Traceability Matrix
SGAM	Smart Grid Architecture Model
SO	System Operator
SUC	System Use Case
SV	State Variable
T&D CP	Transmission and Distribution Coordination Platform
TLS	Traffic Light System
TSO	Transmission System Operator
UC	Use Case
UI	User Interface



List of Figures

Figure 1: WPs interdependencies.....	9
Figure 2: OneNet reference architecture	12
Figure 3: Technical composition of FIWARE TRUE Connector.....	13
Figure 4: Interfaces schema of the OneNet Connector and Middleware.....	13
Figure 5: Functional Requirements: Semantic Development	18
Figure 6: the OneNet Concept Architecture (with the Orchestration Workbench and the Monitoring and Analytics Dashboard).....	21
Figure 7: Components of network traffic monitoring and analysis and their position in OneNet architecture...	22
Figure 8: OneNet Connector Architecture - v1	49
Figure 9: Create Entity - Sequence Diagram	51
Figure 10: Registration - Sequence Diagram	52
Figure 11: Get Entity - Sequence Diagram.....	52
Figure 12: IDS based TRUE Connector [11].....	54
Figure 13: Secure Communication.....	56
Figure 14: Connector Roles and Manifestations.....	56
Figure 15: The three layers of the OneNet web application	58
Figure 16: Sandwich testing strategy for OneNet ecosystem.....	59
Figure 17: Steps to create a test plan	61

List of Tables

Table 2.1: Functional Requirements	14
Table 2.2: OneNet Cross-Platform Services	24
Table 2.3: Cross-platform services to be tested in pilots	26
Table 3.1: Overview of the Integration Plan	31
Table 4.1: Phase A: Deliverables & Milestones.....	34
Table 4.2: Phase A: Integration Planning.....	35
Table 4.3: Phase B: Deliverables & Milestones.....	35
Table 4.4: Phase B: Integration Planning	37
Table 4.5: Phase C: Deliverables & Milestones.....	37
Table 4.6: Phase C: Integration Planning	39
Table 6.1: DPIA Structure.....	48
Table 7.1: OneNet Connector REST APIs	54





Executive Summary

The OneNet Reference Platform (OneNet RP) aims to showcase that the concepts developed and evolved through the various work packages of the OneNet project can result to an architecture and a feasible system that can easily connect any flexibility platform and will enable the various stakeholders to exchange reliable and seamless respective data among them.

Pivotal to this work is the conceptual work that concluded in OneNet's WP5, towards a system design of this OneNet Reference Platform. Main points of this work include:

- A Reference Architecture that combines IDS structure and FIWARE
- A categorisation of Services ("cross-platform services") and Data ("Business Objects") that can serve the specific transactions among stakeholders
- Frameworks and tools to ensure compliance with legal, regulatory and cyber-security requirements of the system.

The first release of the OneNet Reference Platform implements its basic communication capabilities; an easy-to-install connector, common for any stakeholder, deployed in each platform/stakeholder's environment that identifies each participant and enables the exchange of data between them.

OneNet RP does not intend to replace any existing flexibility or TSO-DSO coordination platform. Therefore, the OneNet RP does not "look" into the data exchanged and certainly is not a data repository; OneNet RP is in place to enable the communication and, in case of usage of standardized data profiles (like ENTSO-e's CIM), offers tools to verify the correctness and integrity of such data. In any case participants have the ability to use custom data models, provided that the consumer(s) of such data can cope with them.



1 Introduction

1.1 OneNet scope

The OneNet will create a fully replicable, open, flexible and scalable architecture that enables the whole European electrical system to operate as a single efficient system in which a variety of markets allows the universal participation of stakeholders regardless of their physical location, at every level from small consumer to large producers. Also, by clearly define stakeholder interactions and bringing all possible data exchanges to a European level of harmonization it will fully unlock markets at every level and expand the possibilities for a real commercial exploitation.

The OneNet results will be:

- A data management framework which will support flexibility markets, but also monitor and optimize the overall European electrical infrastructure
- A clear and open architecture that will enable any player to participate at innovative market structures
- A smooth integration of the grid and market operation for TSO and DSO in the innovative market structure
- A new set of customer-centric business models to support next generation service-based markets

According to OneNet Description of Action (DoA), WP6 contributes to the development of the OneNet system, starting from the design of the OneNet Reference Architecture described in D5.2 and all the necessary requirements and specification provided in the other WP5 tasks. In addition, the WP6 will also perform monitoring and evaluation activities during the integration and execution phase, specifically focused on the compliance with the OneNet Reference Architecture and the Cybersecurity aspects.

WPs Interactions

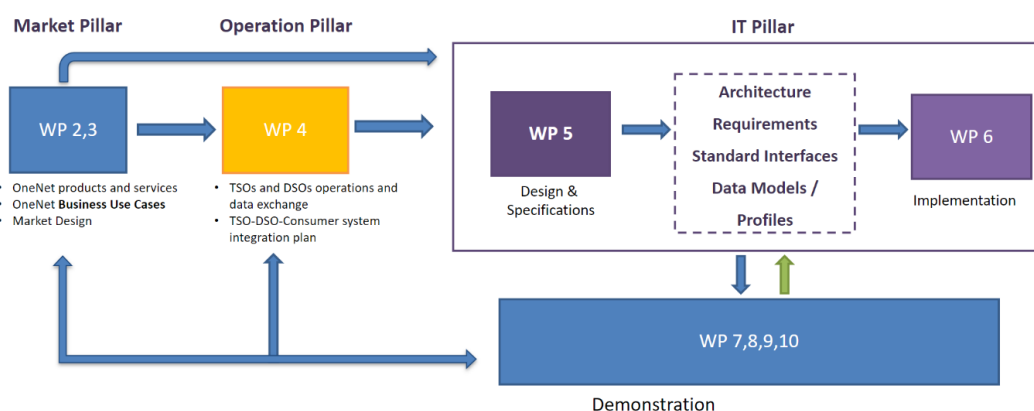


Figure 1: WPs interdependencies



1.2 Report Outline

This report has the following structure:

- **Chapter 2** presents the functional requirements on the OneNet middleware and its components as they will be developed in the tasks within WP6
- **Chapters 3 & 4** describe the integration process of the OneNet middleware with the pilots.
- **Chapter 5** sets a Contingency Plan, i.e. possible risks and their mitigation during the development and integration process
- **Chapter 6** includes a guideline for the pilots towards a Data Processing Impact Assessment (DPIA)
- **Chapter 7** includes the description of the early development of the integrated OneNet Interoperable Network of Platforms by incorporating the described frameworks and components, the interfaces and the communication channels
- **Chapter 8** describes the anticipated testing procedures for the developed S/W, while, last but not least,
- **Chapter 9** refers to the guidelines for the deployment of the OneNet middleware by the pilots.

The ANNEX presents the current version of the GUI Manual.



2 Aligning OneNet Development with other Tasks & Demos

Based on the above, the present integration plan aims to present in a clear manner the methodology applied within Task 6.5 for a successful alignment of individual task components in order to achieve their successful convergence within the overall OneNet Interoperable Network of platforms.

Since the OneNet Interoperable Network of Platforms is comprised of many different frameworks, components, interfaces to external services, systems and data-hubs the Integration plan will act as a roadmap for the overall integration process, focusing on functional requirements and a concise activities time plan.

2.1 Functional Description of OneNet Components

2.1.1 Task 6.1 Decentralized edge-level middleware for scalable platform agnostic data management and exchange

The basic objective of Task 6.1 is to create a decentralized edge-level middleware which will be used as a middleware layer on top of the common IT infrastructure that will enable the exchange of information between all assets and various components that will be integrated in the OneNet Interoperable Network of Platforms. The edge-level middleware will include the following modules:

- Interfaces between involved actors to enable the handling and management of data
- Semantics layer interface to feed this layer with information on the definition of data sharing requirements and how these will be implemented in the components of the system.

Overview of OneNet architectural approach

The analysis of the architectural approach (see *Figure 2*) and structure of OneNet is analytically provided in Deliverable 5.2 [2]. OneNet relies on data interoperability mechanism to all platforms as a matter of supporting data exchange for facilitating market and network operations and the cooperation between network operators, like TSOs and DSOs as well as the involvement of other players like prosumers and aggregators. To achieve the seamless interoperability, fundamental characteristics that shall be covered are adoption of open standards and interfaces, data privacy and data access to regulation for each stakeholder, definition of standard models and protocols for data exchange, provision of data management and dataflow monitoring, identification, authentication, and authorization mechanisms. The decentralized approach and the use of standardized interfaces and mechanisms therefore assume fundamental importance to satisfy all these characteristics and to ensure the necessary scalability for the near real-time data integration and management enabling multi-country and multi-stakeholder near real-time services.

The analysis of IDS reference model and FIWARE interfaces, bring to a hybrid solution using both the standard models for implementing the OneNet Decentralized middleware and the OneNet Connector. The usage of IDS Connector and FIWARE Context Broker was identified as the best solution to be adopted for ensuring a high level of standardization, interoperability, scalability and reuse of OneNet solution.

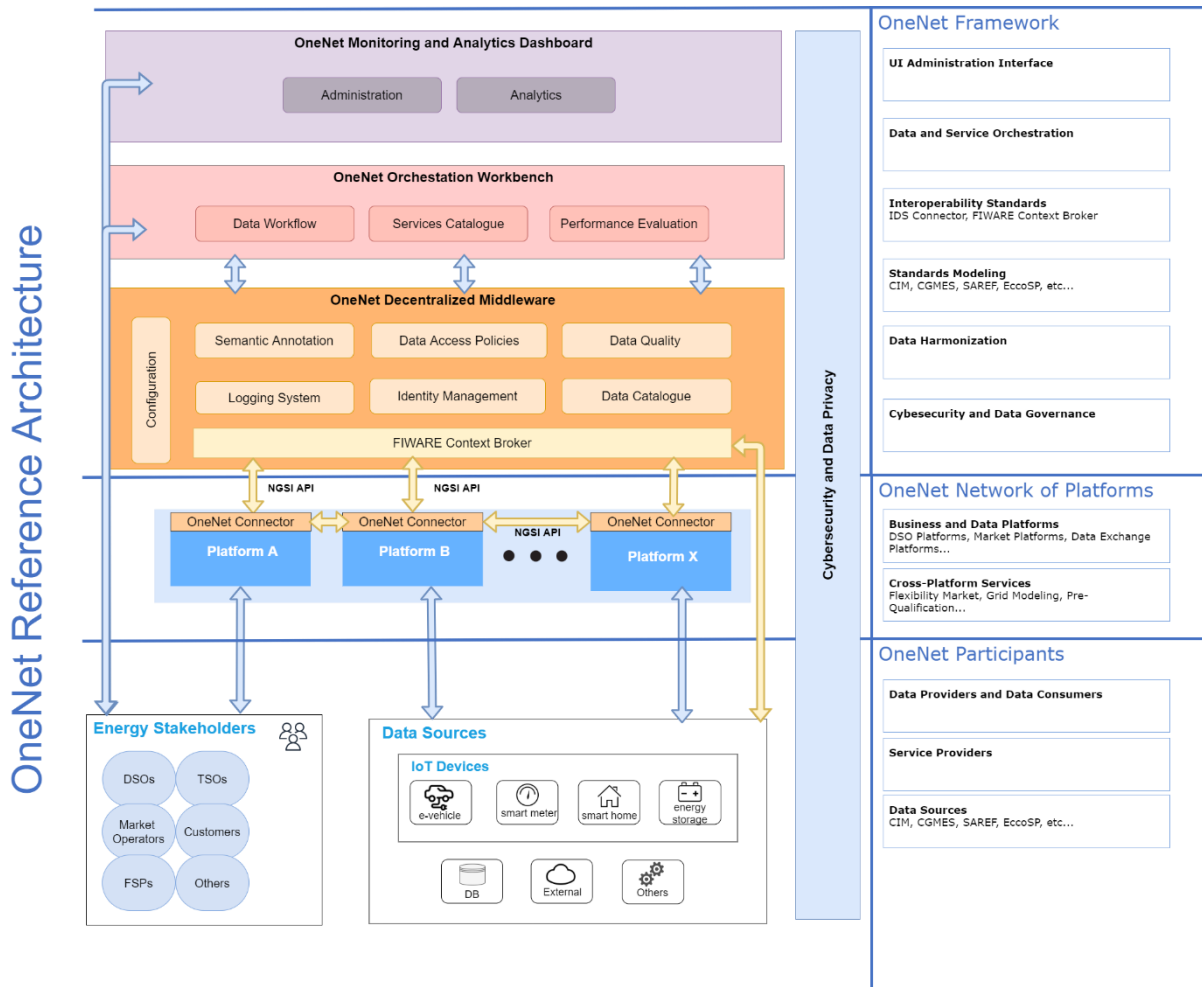


Figure 2: OneNet reference architecture

From the OneNet perspective, one of the most important components is the OneNet Connector, which will be a specific instance of the OneNet Decentralized Middleware and will be placed inside each platform allowing an easy integration and cooperation among the platforms, maintaining the data ownership and preserving access to the data sources.

As described in the D5.4 [4] and D5.5 [5], OneNet connector will rely on the FIWARE TRUE Connector (FTC), a connector for the IDS (International Data Space) ecosystem. FTC enables the trusted data exchange to be active part of an IDS Ecosystem, a virtual data space leveraging existing standards and technologies, as well as governance models well-accepted in the data economy, to facilitate secure and standardized data exchange and data linkage in a trusted business ecosystem. This connector's implementation is compliant with the latest IDS specifications and can be easily customized to fit a wide spread of scenarios thanks to the internal separation of Execution Core Container and Data App. It is integrable with a lot of existing IDS services and totally configurable in terms of internal/external data format (multipart/mixed, multipart/form, http-header) and protocols (HTTP, HTTPS, Web Socket over HTTPS, IDSCPv2). The FTC includes the Execution Core Container, based on the IDS Reference Model for the integration of the IDS based services and metadata exchange as well as the NGSI-LD Data App, able to enable the data exchange using the FIWARE NGSI-LD Context Broker.

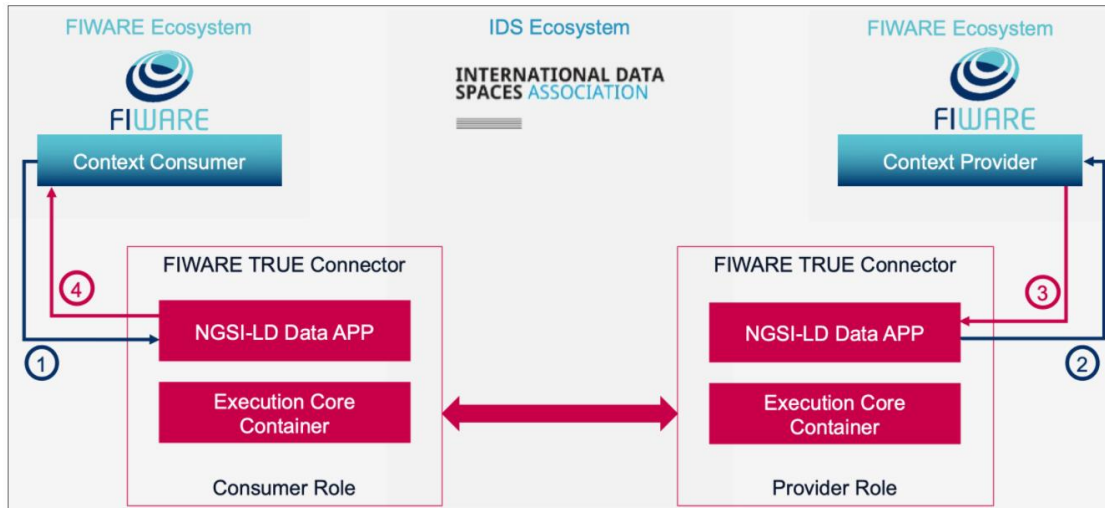


Figure 3: Technical composition of FIWARE TRUE Connector

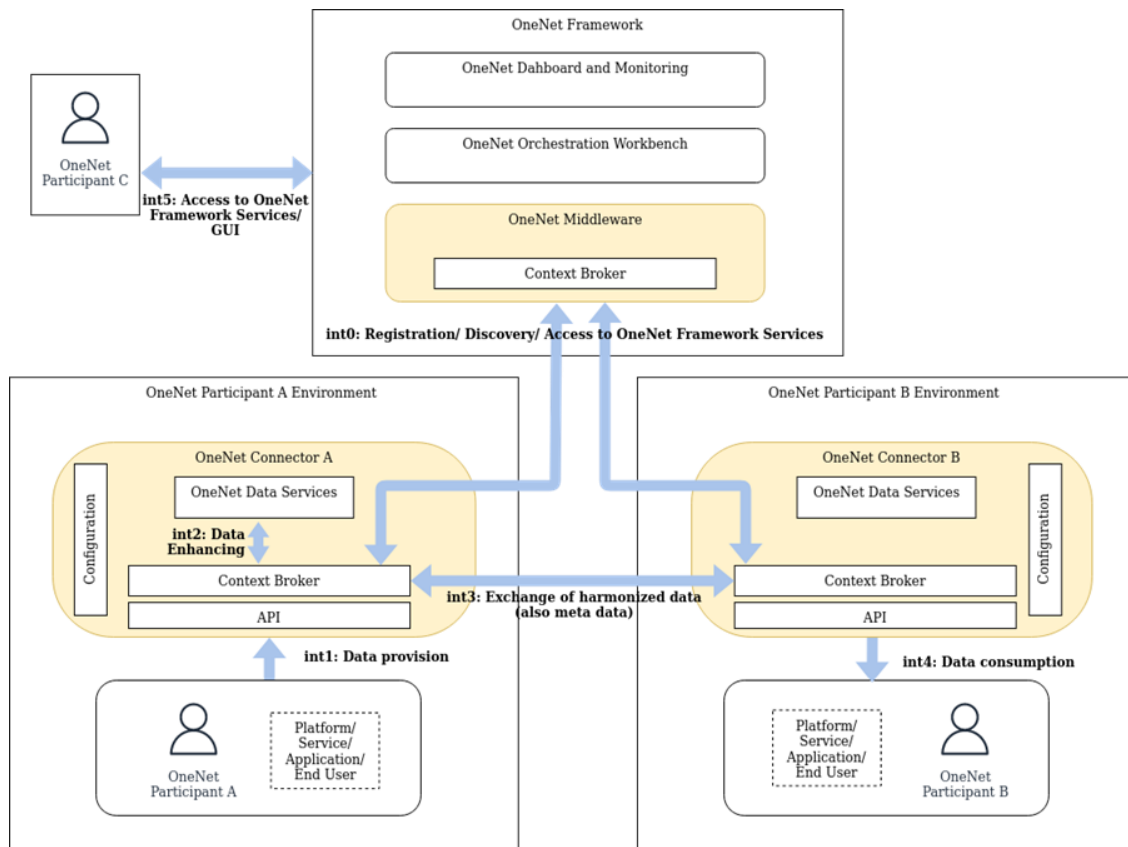


Figure 4: Interfaces schema of the OneNet Connector and Middleware

Functional Requirements

The functional requirements of the edge-level middleware will be based on:

- Demo general and system use cases
- OneNet Reference architecture
- IDS Reference architecture and Data processes
- FIWARE True Connector inherent functionality
- OneNet Cross platform services
- OneNet Interfaces sub-layer
- OneNet Domain agnostic data model (information model)/Semantic Models

All these thematic areas will be analysed to provide a clear and specific set of functional requirements and technical specifications. An example can be shown in the below table:

Table 2.1: Functional Requirements

Requirement ID	Requirement Name	Description	Reference
OneNet_FUR_01	The OneNet system must enable exposure of list of data/ services from vertical WPs to third parties	System of the OneNet Participant has many features/roles and data. Those can be accessed through API's by third party. The list of services/data and their properties can be retrieved automatically by special API - "Catalogue service". This list can be provided by API to OneNet system, in order to be exposed to potential third parties.	Northern Cluster, General SUC_01, General SUC_02
OneNet_FUR_02	The OneNet system must enable role-based access for data/service to authenticated users.	Every data/service responds to authenticated requests only. In case third party need the access then the authentication/secure channel needs to be established.	Northern Cluster
OneNet_FUR_03	The OneNet system must visualize and provide analysis tool for activity logs.	User activity trace logs, technical performance or problem related logs are generated and could be exposed from demonstrator's	Northern Cluster, GSUC_01

		implementation to third parties, through the OneNet system	
OneNet_FUR_04	The OneNet system must facilitate the communication of the SO's flexibility needs to external interested stakeholders	The SOs, i.e., DSO and TSO, shall be able to make available to stakeholders their flexibility needs in different timeframes, e.g., Day-ahead and Intra-day, through the utilization of the OneNet system.	DSUC_WE_PT_02
OneNet_FUR_05	The OneNet system must facilitate market results to be disseminated to external interested stakeholders	The local market platform publishes collected market results through OneNet system to external interested parties.	DSUC_SP_01
OneNet_FUR_06	The OneNet system must facilitate data exchange amongst SOs, MOs, and FSPs participating in the market-based flexibility procurement process, for prequalification, market clearing, evaluation and real-time control purposes.	Prequalified limits in the interface between the HV/MV (TSO) and MV/LV (DSO) that FSPs exist are sent to the market (TSO market or local DSO market) in order to be taken into consideration by the market operator in the allocation of the awarded bids to the FSPs. In addition, MO publishes the awarded bids to the operators through the OneNet. After the activation of the flexibility, evaluation report of the FSP's performance is sent to the market operator through the OneNet platform. Finally, communication between the DSO control center (ABCM-D platform is developed in the context of Cypriot Demonstrator) and the local FSPs connected to the distribution grid through the OneNet system	DSUC_SO_CY_02, DSUC_SO_CY_03, DSUC_SO_CY_04
OneNet_FUR_07	The OneNet system must connect the involved in the Demo parties to external actors responsible for TSO-DSO coordination	TSO/DSO coordination process takes place through the utilization of OneNet. This specifically includes: DSO demand finalization, flexibility registration, bid prequalification and market result broadcasting	DSUC_EA_HU_01 DSUC_EA_HU_02 DSUC_EA_HU_03

OneNet_FUR_08	OneNet system must be able to manage and certificate the identity of each OneNet Participant	OneNet system manage the identities of all the OneNet participants offering an Identity Provider	GSUC_01 GSUC_03
OneNet_FUR_09	OneNet system must be able to register/unregister a OneNet connector	OneNet Connector need to register itself before starting any data exchange process	
OneNet_FUR_10	Each OneNet Participant must be uniquely identified using certification	OneNet Participants are uniquely identified within the OneNet ecosystem, using certification process and establishing trust among all participants.	
OneNet_FUR_11	Each OneNet Connector have a unique certificate and identifier		
OneNet_FUR_12	Each OneNet Connector is able to verify the identity of the other OneNet Connectors		
OneNet_FUR_13	OneNet participant must be able to run the OneNet connector in its own environment	OneNet Middleware leverage on the IDS decentralized approach. The OneNet Connector provided by OneNet must be deployable in any environment	
OneNet_FUR_14	The OneNet Participant must be able to configure its own OneNet Connector	OneNet connectors are configurable by the OneNet participants using specific interfaces	
OneNet_FUR_15	The OneNet connector must be able to send metadata of a data source to one or more Brokers	Once the connector is configured it is able to connect the Brokers for starting data exchange. The connector is able to provide and/or search metadata as well as discover for new data sources and participants.	
OneNet_FUR_16	The OneNet Participant must be able to search and discover other OneNet Participants		
OneNet_FUR_17	The OneNet Connector must be able to search for metadata connecting to a Broker		
OneNet_FUR_18	The OneNet Connector must be able to exchange data with other connectors using pull and/or push mechanisms	The data exchange process happens end-to-end exploiting pull or push mechanisms.	
OneNet_FUR_19	The OneNet system must be able to support the creation, management and usage of vocabularies	A feature provided by OneNet system is the Vocabulary Provider. It	

OneNet_FUR_20	The OneNet participant could use vocabularies for creating and structuring its metadata	manages and offers vocabularies (i.e., ontologies, reference data models, or metadata elements) that can be used to annotate and describe datasets.	
OneNet_FUR_21	The OneNet system should offer data services/apps for data processing and transformation	One of the main features of the OneNet system is the possibility to enrich, transform, validate and harmonize the data processed. In addition, the OneNet allow to log all the data transaction.	
OneNet_FUR_22	The OneNet system should be able to log any data transaction between any OneNet participant		
OneNet_FUR_23	The OneNet system should be able to assess the quality of data processed		
OneNet_FUR_24	The OneNet system should be able to perform a semantic validation of the data processed		
OneNet_FUR_25	The OneNet system could use AI mechanism for empowering Data services	For improving the Data Services offered by the OneNet system, some AI mechanism could be implemented.	GSUC_03
OneNet_FUR_26	The OneNet system should be able to integrate any kind of data sources using Context Broker	The usage of the FIWARE context broker could facilitate the integration of any kind of data source, using a standard API based approach.	

Methodological Semantic Development Approach

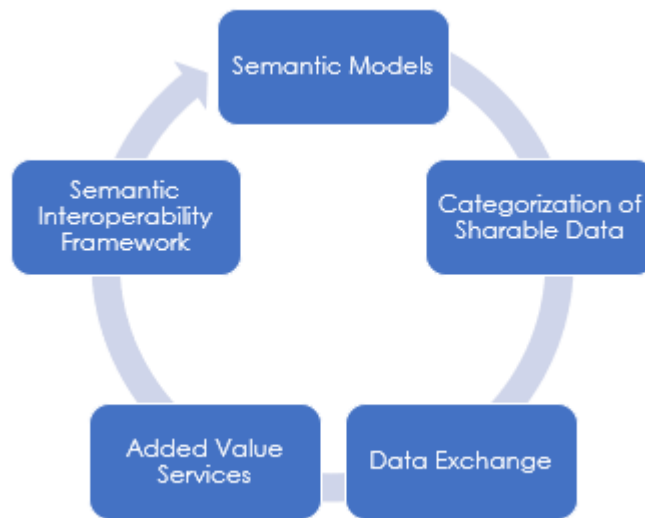


Figure 5: Functional Requirements: Semantic Development

1. **Definition and development of Semantic Data Models:** The semantic data models will assist the IT systems of the stakeholders with the interpretation of the exchanged information between organizations and to invoke actions.
2. **Categorization of sharable data:** Categorization of the sharable data by each authority will take place, classifying them in unstructured (text data), semi-structured (e.g. XML/JSON normalized data) and structured (e.g. data in the operators' Data Hubs and Energy Data Management Systems).
3. **Data exchange:** The semantic data models will be developed based on a data-driven approach in order to lead to an improved data exchange.
4. **Added value services:** In parallel specific added value services will be developed to support and build upon the semantic models such as quality checking, data consolidation and aggregation.
5. **Generation of a semantic interoperability framework:** A framework for orchestrating semantic interoperability e.g. based on the International Data Spaces for data sharing will be generated to describe the data exchange interfaces for read/write access and import/export of available data and output analytics.

- *Quality of Service: Low-level tasks*

The development of the edge level middleware will take into consideration the following quality of service properties:

Performance	The measurement of response time and throughput with respect to user load conditions.
Availability	A measure of how often a system's resources and services are accessible to end users, often expressed as the uptime of a system.
Scalability	The ability to add capacity (and users) to a deployed system over time. Scalability typically involves adding resources to the system but should not require changes to the deployment architecture.
Security	A complex combination of factors that describe the integrity of a system and its users. Security includes authentication and authorization of users, security of data, and secure access to a deployed system.
Latent capacity	The ability of a system to handle unusual peak loads without additional resources. Latent capacity is a factor in availability, performance, and scalability qualities.
Serviceability	The ease by which a deployed system can be maintained, including monitoring the system, repairing problems that arise, and upgrading hardware and software components.

2.1.2 Task 6.2 Cross stakeholder Governance for Energy Data Exchange

Data exchange governance layer of OneNet solution builds on regional and national initiatives, including architectures, data exchange platforms and frameworks like BRIDGE reference architecture DERA, OPEN DEI, ETIP SNET, GAIA-X, IDSA, European Interoperability Framework, ECCo SP, etc. Recommendations given in the European Smart Grids Task Force Expert Group 1 report should also be followed to facilitate interoperability by using available standards as a basis like IEC CIM model, by relying on role models like HEMRM and by applying technology-neutral business requirements. Based on the review of existing data exchange initiatives, platforms, frameworks, and recommendations governance aspects will be identified, categorised and benchmarked.

The governance model must recognize the variety of different platforms and systems, fit to different market designs and business processes, enable cross-stakeholder, cross-border and cross-sector data exchanges, ensure easy access to data satisfying GDPR requirements, facilitate TSO-DSO coordination from customer perspective, ensure scalability through open-source principle and agreed rules. Conceptual overview of cross stakeholder governance will include services/processes/functionalities required to fit different market designs and business processes enabling cross-stakeholder, cross-border, and cross-sector data exchanges.

OneNet architecture and requirements will be analysed with focus on data governance. Other OneNet results related to data governance (data quality, cybersecurity, data standards, etc.) will also be reviewed. Based on this data governance aspects for OneNet middleware can be defined.

Data governance model based on BRIDGE reference architecture will be developed. This includes elaboration of governance requirements: principles, policies, tools. Governance Requirements Traceability Matrix (GRTM) vis-a-vis respective test cases will be created.

2.1.3 Task 6.3 Data and Service Interoperability, Integration with FIWARE, and Homogenization Management

The main goal of the Task 6.3 is the development of the Data Integration & Homogenization sub-layer as part of the OneNet Decentralised Middleware to:

- manage the access to data in an interoperable manner
- provide standardized data gates relying on existing or promising standardized data models by evolving the FIWARE Context Broker NGSI into NGSI-LD.
- Implement interfaces and harmonisation for relevant IEC CIM standardized data models (e.g., IEC 62325-301)

The main outputs of the task consist of the integration of FIWARE Context Broker and implementation of NGSI-LD API within the OneNet Connector and Decentralised Middleware and the implementation of data services for the integration and harmonisation of XML/RDF CIM standardised data models into JSON-LD.

2.1.4 Task 6.4 AI, Big Data, IoT Workbench to support scalable near real time big data-enabled services

Task 6.4 foresees the implementation of two main components: the OneNet Orchestration Workbench and the OneNet Monitoring and Analytics Dashboard.

OneNet Orchestration Workbench is the component able to orchestrate and evaluate the performance and scalability of the cross-platform services that will be integrated and implemented in the OneNet System.

Any OneNet Participant will be able to test and evaluate its own service exploiting the OneNet Orchestration Workbench, that allows to integrate data coming from the OneNet Middleware and to implement a data pipeline orchestration.

From a functional perspective the OneNet Orchestration Workbench will support the integration and the evaluation of the performance and scalability of the AI, IoT and Big Data cross-platform services for market and grid operations.

The OneNet Orchestration Workbench will allow to integrate data coming from the OneNet middleware and will allow the interaction with the OneNet Participants which will be able to deploy and evaluate their own services, integrating it with analytics and data visualization features.

OneNet Monitoring and Analytics Dashboard is the principal Graphical User Interface of the OneNet Framework. It allows the access to the OneNet Participants for monitoring and data analytics features, as well as for the OneNet Administrators for configuration and administration tools.

Main features of the OneNet Monitoring and Analytics Dashboard should be:

- administrative and configuration tools;
- easy integration with the OneNet Orchestration Workbench and OneNet Middleware;
- data-analytics dashboard;
- monitoring and alerting dashboard for data processes and platform integrations;
- user-friendly selection of data sources and services from the catalogues.

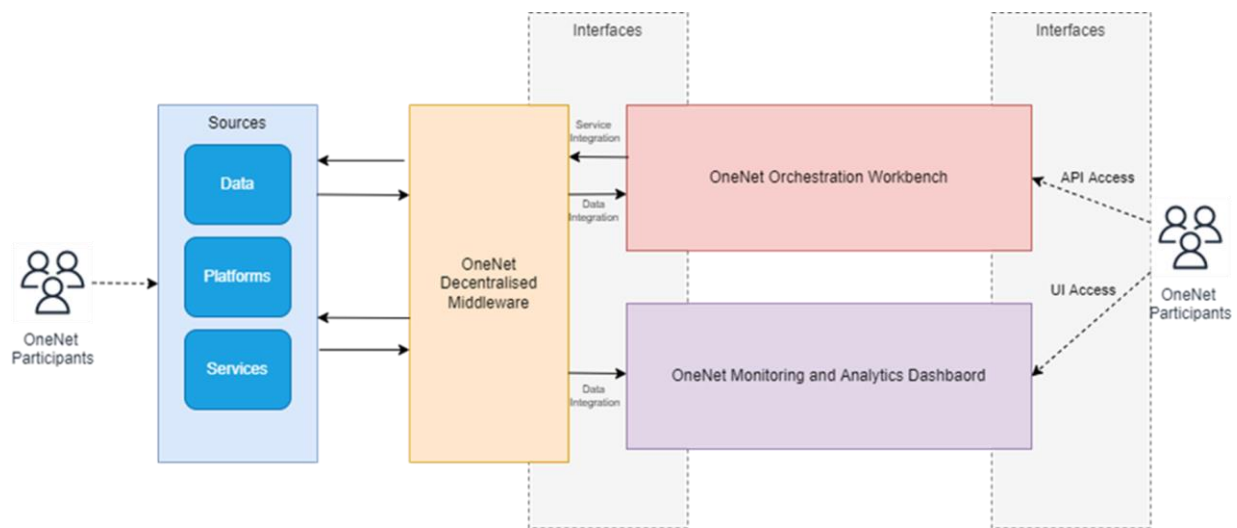


Figure 6: the OneNet Concept Architecture (with the Orchestration Workbench and the Monitoring and Analytics Dashboard)

2.1.5 Task 6.6 Legal, Regulatory, Privacy and Cybersecurity Management & Compliance Tools

The basic objectives of task 6.6 cover the delivery of two main components:

Network Traffic & Endpoint Infrastructure Monitoring

This tool is responsible for continuous monitoring of the source traffic/logs/events that come through the OneNet Connector, in an effort to assist on the cyber-security preservation aspects of the OneNet solution.

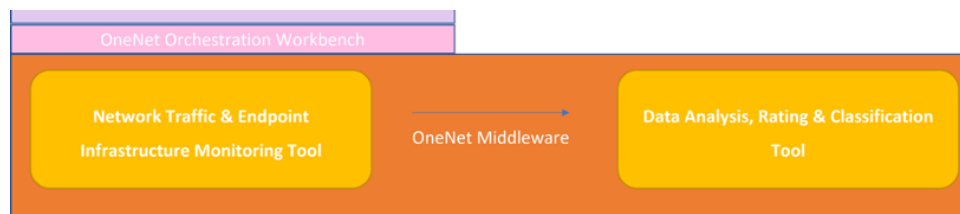
Malicious network activity and system vulnerabilities can be identified so that data access policies to the OneNet system can be updated or enhanced.

Data Analysis, Rating & Classification

This tool is responsible for network traffic classification or clustering based on the machine learning algorithm used (supervised or unsupervised). The algorithm can extract useful features around the data traffic such as basic features (source/destination IP address, source/destination host port, frame length), time-based features (number of frames received in a specific time interval), connection-based features (number of packets flowing from source to destination and vice versa) or even classify under normal/abnormal traffic.

Figure 7 shows the position of these two components in the OneNet architecture and how these tools interact with each other. When an OneNet participant performs a rest api request, by interacting with the Dashboard frontend, this request represents an incoming web log event to the OneNet backend. The **Network Traffic & Endpoint Infrastructure Monitoring** tool acts as a real-time network packet analyzer that can be used to provide an *application monitoring and performance analytics system*. It works by capturing the network traffic between application servers, decoding the application layer protocols (HTTP, MySQL, Redis, and so on), correlating the requests with the responses, and recording the interesting [fields](#) for each transaction. The

Figure 7: Components of network traffic monitoring and analysis and their position in OneNet architecture



correlated transactions are inserted directly into a central storage solution for further processing. This solution is extensible also for the logs analysis, where the system logs can be centrally harvested and stored for further analysis.

On the other hand, the **Data Analysis, Rating & Classification tool performs the analysis of the stored web log events, the extraction of useful features regarding incoming traffic, the classification of incoming traffic to sub-categories and can be extended to also detect potential anomalies in the network. For the anomaly detection we will use the Isolation Forest methodology.** Isolation Forests were built based on the fact that anomalies are the data points that are “few and different”.

The results of the aforementioned tools will be used to feed the Data Access Policies Component. In case a user request is identified as “anomalous” by the tools then the data access policies of the corresponding user will be updated accordingly i.e. he might be banned from accessing specific data sets.

The proposed minimal technical stack for the implementation of the tools is:

- Django framework with respective python libraries
- Angular framework
- Elastic stack with PacketBeat and FileBeat.

2.1.6 Task 6.7 Tools evolution management and Compliance to Reference Architecture

This Task will ensure the quality and compliance with OneNet Reference Architecture of all the tools developed within the project that will be integrated in the OneNet Network of Platforms. This process will follow an iterative and incremental approach, ensuring a continuous feedback loop with demonstrators.

The tools to be evaluated have been already identified and provided in D6.7. Below the complete list:

- Context Broker
- Vocabulary provider, Semantic Tools and Services
- Identity Provider and Management (Authentication & Authorization)
- Data Access Control (Usage Control)
- Clearing House
- Data Quality Tool
- User Interface (OneNet Connector)
- Data Process Management (Orchestration & Workflow) Module
- Service Management Module
- Performance Evaluation Tool
- Administration Dashboard
- Analytics & KPIs Dashboard
- Network Traffic & Endpoint Infrastructure Monitoring Tool
- Data Analysis, Rating & Classification Tool

The next activities of the T6.7 will focus on monitoring, evaluation, and result phase for each release of the components of the OneNet Framework.

2.2 SUCs / Services to be tested in each Demo

The OneNet demonstrations will integrate the OneNet System using the cross-platform services as listed in Table 2.2. For some demonstrations, the instance that receives data through the OneNet System is not a part of the demo itself but any eligible OneNet Participant (see rightmost column of the table). More detailed information on the SUCs, cross-platform services to be tested, the planned data exchanges through the OneNet System, and the involved actors per demo are available in Table 2.3.

Table 2.2: OneNet Cross-Platform Services

Demo		OneNet service(s) used by demo	Data receiver part of demo (Part of demo cluster for regional use case)
Northern Cluster		OneNet_04REIN_0003 Verification process results OneNet_05MRKT_0001 Market bid/offer OneNet_05MRKT_0002 Market clearing/results OneNet_05MRKT_0003 Required resources OneNet_05MRKT_0004 Market information OneNet_05MRKT_0006 Resource Optimization OneNet_05MRKT_0010 Market opening/closing OneNet_08REQU_0002 Qualification results OneNet_08REQU_0003 Requalification results	Yes
Eastern Cluster	Czech Republic	OneNet_02MEMO_0004 Grid state OneNet_03FORC_0004 Maintenance forecast OneNet_05MRKT_0002 Market clearing/results OneNet_05MRKT_0003 Required resources	No
	Hungary	[pending]	
	Poland	OneNet_05MRKT_0002 Market clearing/results OneNet_05MRKT_0004 Market information	No

	Slovenia	OneNet_05MRKT_0002 Market clearing/results	No
	Regional	OneNet_05MRKT_0004 Market information	Yes
Southern Cluster	Cyprus	OneNet_04REIN_0001 Energy Account Document OneNet_05MRKT_0002 Market clearing/ results OneNet_05MRKT_0003 Required resources OneNet_10RECO_0002 Execution Orders	Yes
	Greece	OneNet_03FORC_0001 Forecast data (general) OneNet_05MRKT_0008 Congestion data	No Part of Cypriot demo
	Regional	OneNet_02MEMO_0005 Cyber event OneNet_03FORC_0001 Forecast data (general)	Yes
Western Cluster	France	OneNet_09SSAC_0001 Activation document	No
	Portugal	OneNet_05MRKT_0003 Required resources	Yes
	Spain	OneNet_05MRKT_0002 Market clearing/ results	Yes
	Regional	OneNet_08REQU_0002 Qualification results OneNet_08REQU_0006 Request for qualification	Yes

Table 2.3: Cross-platform services to be tested in pilots

OneNet ID (T5.3) and Service Name (T5.3)	Demo(s) using this service	Name of use case(s) using this service	Short description of data exchange in demo	Data Sender in demo	Data Receiver in demo
OneNet_02MEMO_0004 Grid State	Czech Republic	Non-frequency services	Provide information on sudden outages from Czech AccessNet platform to eligible OneNet Participant	Czech AccessNet platform	Eligible OneNet Participant
OneNet_02MEMO_0005 Cyber event	Southern Regional	Regional critical infrastructure security awareness information exchange	Exchange information about cyber-attack or potential cyber threat between ICT departments of GR and CY TSOs and DSOs	ICT department of Greek/ Cypriot TSOs and DSOs	ICT department of Greek/ Cypriot TSOs and DSOs
OneNet_03FORC_0001 Forecast data (general)	Greece	Greece TSO/ DSO Flexibility Platform Data and Services – Gateway to the OneNet System	Communicate critical (scenario #1) or required (scenario #2) information on weather forecast or occurrences, load or generation forecast or occurrences from Greek TSO/DSO Coordination platform to Cypriot TSO/DSO	Greek TSO/DSO coordination platform	Cypriot TSO/DSO coordination platform
	Southern Regional	Regional critical infrastructure security awareness information exchange	Communicate incoming severe weather conditions (storms and icing) between GR and CY TSOs and DSOs	GR and CY TSOs and DSOs	GR and CY TSOs and DSOs
OneNet_03FORC_0004 Maintenance forecast	Czech Republic	Non-frequency services	Provide information on planned outages (location, duration) from Czech AccessNet platform to eligible OneNet Participant	Czech AccessNet platform	Eligible OneNet Participant
OneNet_04REIN_0001 Energy Account Document	Cyprus	Evaluation of the Flexible Service Providers response	Communicate report on FSPs' response to awarded bids from TSO/ DSO to MO and FSPs	TSO, DSO	MO, FSPs
OneNet_04REIN_0003 Verification process results	Northern Cluster	Flexibility verification and settlement	Communicate verification results from FR to MO and FSP	FR	MO, FSP

OneNet_05MRKT_0001 Market bid/ offer	Northern Cluster	Grid Qualification of Resource and Bid Optimization and Procurement	Collect/forward information about flexibility bids send relevant bids to European platform	MO, TSO/ DSO coordination platform	TSO/ DSO coordination platform, MO, EU MO
OneNet_05MRKT_0002 Market clearing/ results	Northern Cluster	Procurement and activation support	Send and receive market outcome from MO to FR, T&D CP, SO	MO	FR, TSO/DSO coordination platform, SO
	Spain	Local Market platform - Market Session	Send the collected market results to the OneNet system to make them available from the LMP (Local Market Platform) to neighboring SOs	Local Market Platform	Neighboring SOs
	Cyprus	Evaluation of the Flexible Service Providers response & Coordination of the distributed flexible resources	Communicate awarded bids from MO to FSPs, DSO and TSO & communicate cleared awarded bids from MO to DSO and FSPs	MO	FSOs, DSO, TSO
	Slovenia	Congestion management in distribution grids under market conditions - Bidding	Provide access to trading results of Flexibility Market Platform (UI) to any eligible OneNet Participant	Flexibility Market Platform (UI)	Eligible OneNet Participant
	Poland	Bidding for day-ahead balancing services and Bidding for congestion management (CM) and voltage control (VC) services	Provide access to market results of Polish Flexibility Platform to any eligible OneNet Participant	Flexibility Platform	Eligible OneNet Participant
	Czech Republic	Non-frequency services	Provide access to information on provided flexibility services (market results) from Czech AccessNet platform to eligible OneNet Participant	AccessNet platform	Eligible OneNet Participant
OneNet_05MRKT_0003 Required resources	Northern Cluster	Bid optimization	Send adjusted balancing purchase offers to European platform	TSO/DSO coordination platform	European platform
	Portugal	Day-ahead & Intraday Flexibility needs for DSO/TSO within OneNet System	Make available the amount of flexibility needed to solve constraints per node/ zone for the next day or for the next 6 hours in OneNet System	DSO plan/ forecast system	DEP of TSO and DSO

	Cyprus	Prequalification of the location-based limit of each market product	Communicate limits/ margins of TSO + DSO grids to the Market Operator	TSO, DSO	Market operator
	Czech Republic	Non-frequency services	Provide access to information on required flexibility services (required resources) from Czech AccessNet platform to eligible OneNet Participant	AccessNet Platform	Eligible OneNet Participant
OneNet_05MRKT_0004 Market information	Northern Cluster	Preparation for flexibility trading, Procurement and activation support, Procurement	Exchange product specification including qualification, verification of FR to MO, Confirm/forward product registration of FR to MO, Check/forward FSP prequalification per product from FR/MO to MO/FR, forward list of FSPs of MO to T&D CP, forward new contract holder of FR to MO, get list of FSPs to FR	MO, FR	FR, MO, TSO/ DSO coordination platform
	Eastern Regional	Flexibility Market data aggregation	Request and exchange aggregated flexibility market data from national market aggregator to eligible OneNet Participant	National market aggregator	Eligible OneNet Participant
	Poland	Bidding for day-ahead balancing services and Bidding for congestion management (CM) and voltage control (VC) services	Provide access to information about the market (e.g., object, participants) from Polish Flexibility Platform to any eligible OneNet Participant	Flexibility Platform	Eligible OneNet Participant
OneNet_05MRKT_0006 Resource optimization	Northern Cluster	Bid Optimization	Report and forward optimization results	TSO/ DSO coordination platform	MO
OneNet_05MRKT_0008 Congestion data	Greece	Greece TSO/ DSO Flexibility Platform Data and Services – Gateway to the OneNet System	Communicate the forecast or occurrences of network congestions (scenario #1 and #2) from Greek TSO/DSO Coordination platform to Cypriot TSO/DSO	Greek TSO/DSO coordination platform	Cypriot TSO/DSO coordination platform

OneNet_05MRKT_0010 Market opening/ closing	Northern Cluster	Procurement and Flexibility call for trading opening	Open/Close Market Session according to the call tender	T&D CP, MO	T&D CP, MO
OneNet_08REQU_0002 Qualification results (for new resource group registration)	Northern Cluster	Procurement and activation support & Procurement	Request/forward prequalification of FSP, product and grid from FR to MO, SO and T&D CP & Check prequalification status	FR	MO, SO, T&D CP
	Western Regional	Cluster preparatory Phase - cross-SO grid pre-qualification	Make available the accepted/ rejected registration pre-qualification from a foreign MO to national MO	Foreign MO	National MO
OneNet_08REQU_0003 Requalification results	Northern Cluster	Preparation for Flexibility trading	Forward/receive prequalification of FSP, product and grid from FR/MO to MO/FR	FR, MO	FR, MO
OneNet_08REQU_0006 Request for qualification	Western Regional	Cluster preparatory Phase: cross-SO grid pre-qualification	Send the fulfilled pre-qualification requirements from a national MO to a foreign MO, make available the accepted/rejected registration pre-qualification from a foreign MO to a national MO	National MO, Foreign MO	Foreign MO, national MO
OneNet_09SSAC_0001 Activation document	France	STAR platform – TSO automated activation	Send the TSO order from STAR platform to eligible OneNet Participant (3rd Party)	STAR Platform	Eligible OneNet Participant

OneNet_10RECO_0002 Execution orders	Cyprus	Coordination of distributed flexible resources	The coordination signal will be sent to the FSPs for coordinating the response of FSPs. These signals can be called execution order signals for the provision of flexibility service. In this signal the value of special product needed for particular flexibility service will be included.	DSO	FSPs
----------------------------------------	--------	------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----	------

3 Overview of the Integration Plan

Table 3.1: Overview of the Integration Plan

Phase	Name	Task	Starts	Ends
A	First development phase (Decentralized edge-level middleware for scalable platform agnostic data management and exchange Early Prototype)			31/07/22
A.1	General Functional Requirements Definition - First version			31/05/22
A.1.1	Middleware Functional Requirements definition & review			16/05/22
A.1.2	Assessment of demo specific functionalities (common UCs for data exchange)			16/05/22
A.1.3	Analysis of requirements on Data Integration & Homogenization			31/05/22
A.1.4	Analysis of requirements for OneNet Orchestration Workbench			31/05/22
A.1.5	Analysis of requirements for OneNet Monitoring and Analytics Dashboard			31/05/22
A.2	Middleware development - First version	6.1 & 6.3		31/07/22
A.2.1	Base connector implementation			31/07/22
A.2.2	Interfaces & APIs			31/07/22
A.2.3	FIWARE Context Broker NGSI-LD integration			31/07/22
A.2.4	Mapping Data Models into NGSI-LD - First version			31/07/22
A.2.5	Integration and Homogenization sub-layer implementation - First version			31/07/22
A.2.6	User Interface base version			31/07/22
MS12	Decentralized edge-level middleware for scalable platform agnostic data management and exchange Early Prototype			31/07/22
D6.5	OneNet Reference Platform First Release			31/07/22
	OneNet 1st release (OneNet Reference Platform, First Version)			31/07/22
A.3	Integration and Testing -1-	6.5	01/09/22	30/09/22
A.3.1	OneNet Deployment Guide [Integration Guide update]			31/07/22
A.3.2	Initial Lab Testing			30/09/22
A.3.3	Ensure functional requirements compliance			30/09/22
B	Second development phase (OneNet Reference Platform, Intermediate Version)			31/01/23
B.1	General Functional Requirements Definition final version			30/09/22
B.1.1	Functional requirements middleware evolution			30/09/22
B.1.2	Workbench & Dashboard final functional requirements			30/09/22
B.1.3	Integration and Homogenization final functional requirements			30/09/22
B.1.4	Legal, Regulatory, Privacy and Cybersecurity Management & Compliance Tools final requirements			30/09/22
B.1.5	Definition of Demo specific UCs			30/09/22
B.1.6	Concluding DPIA			30/09/22
B.1.7	Concluding Contingency plan & Risk management			30/09/22
B.2	Middleware development - Intermediate version	6.1 & 6.3		31/01/23
B.2.1	Connector implementation			31/01/23
B.2.2	Interfaces & APIs			31/01/23
B.2.3	Mapping Data Models into NGSI-LD - Final version			31/01/23
B.2.4	User Interface finalisation			31/01/23

B.2.5	Authentication & Authorisation base version			31/01/23
B.2.6	Integration and Homogenization sub-layer implementation - Final version			31/01/23
B.2.7	Data quality & harmonisation services			31/01/23
B.3	Workbench & Dashboard development	6.4		31/01/23
B.3.1	Orchestration workbench implementation			30/11/22
B.3.2	Monitoring and Analytics Dashboard implementation			30/11/22
B.3.3	Integration with OneNet framework			31/01/23
B.4	Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance first version	6.6		31/01/23
B.4.1	Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance first version implementation			31/01/23
B.4.2	Integration with middleware			31/01/23
B.4.3	Ensure NFR compliance			31/01/23
B.5	Ensure reference architecture compliance (first evaluation)	6.7		31/01/23
MS16	OneNet Reference Platform, Intermediate Version			31/01/23
D6.1	Report on decentralized edge-level middleware for scalable platform agnostic data management and exchange			31/01/23
	OneNet 2nd release (OneNet Reference Platform, Intermediate Version)			31/01/23
B.6	Integration and Testing -2-	6.5		30/03/23
B.6.1	Testing individual components			28/02/23
B.6.2	Testing APIs & Interfaces			28/02/23
B.6.3	Integration of all components			28/02/23
B.6.4	Integrated platform lab testing			28/02/23
B.6.5	Integrated platform field testing			30/03/23
B.6.6	Intermediate OneNet prototype demo testing			30/03/23
C	Final development phase (OneNet Reference Platform, Final Version)			31/07/23
C.1	Functional & Non-Functional requirements finalization			31/07/23
C.2	Middleware development - Final version	6.1 & 6.3		31/07/23
C.3	Workbench & Dashboard development - Final version	6.4		31/07/23
C.4	Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance - Final version	6.6		31/07/23
C.5	Ensure reference architecture compliance (second evaluation)	6.7		30/03/23
C.6	Final Assessment			31/07/23
MS13	Updated Report on Cross stakeholder decentralized Data Governance for System Data Exchange			30/03/23
D6.2	Report on Cross Stakeholder decentralized Data Governance for System Data Exchange			30/03/23
D6.3	Report on Extended Interoperability and Management with FIWARE			30/03/23
D6.4	Report on AI, Big Data, IoT Orchestration Workbench			30/03/23
MS14	Extended Interoperability Management with FIWARE Final Version			31/05/23
MS15	AI, Big Data, IoT Orchestration Workbench Final Version			31/05/23
MS17	OneNet Reference Platform, Final Version			31/07/23
MS18	Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance Final Version			31/07/23

MS19	Tools for evolution management and Compliance to Reference Architecture			31/07/23
D6.6	Report on Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance			31/07/23
D6.8	Report on OneNet Reference Platform and Components Final Release			31/07/23
C	OneNet final release (OneNet Reference Platform and Components Final Release)			31/07/23
C.7	Integration and Testing -3-	6.5		
C.7.1	Testing individual components			30/06/23
C.7.2	Testing APIs & Interfaces			30/06/23
C.7.3	Integration of all components			30/06/23
C.7.4	Integrated platform lab testing			31/07/23
C.7.5	Integrated platform field testing			31/07/23
C.7.6	Final OneNet prototype testing			31/07/23
C.7.7	OneNet Validation (in conjunction with C.6)			31/07/23
C.7.8	WP7 Demos implementation			30/09/23
C.7.9	WP8 Demos implementation			30/09/23
C.7.10	WP9 Demos implementation			30/09/23
C.7.11	WP10 Demos implementation			30/09/23
C.7.12	OneNet integrated platform validation			30/09/23
C.8	Final Reporting			30/09/23

4 Detailed Planning towards Integration

The implementation efforts along with the integration process will be split in three distinct development phases according the project time-plan.

4.1 Phase A: First Development Phase (Decentralized edge-level middleware for scalable platform agnostic data management and exchange early prototype)

Table 4.1: Phase A: Deliverables & Milestones

Deliverables & Milestones		
MS12	Decentralized edge-level middleware for scalable platform agnostic data management and exchange Early Prototype	31/07/22
D6.5	OneNet Reference Platform First Release	31/07/22

4.1.1 General Functional Requirements Definition - First version

During this stage the OneNet framework preliminary functional requirements will be defined forming the basis for the development effort scope. The initial set of requirements will focus on the following:

- Middleware Functional Requirements definition & review
- Assessment of demo specific functionalities
- Analysis of requirements on Data integration & homogenization
- Analysis of requirements for OneNet orchestration workbench
- Analysis of requirements for OneNet Monitoring & Analytics dashboard

As a general rule the functional requirements will be based on the analysis of:

- Business Rules
- Data exchange processes, adjustments and cancellations
- Administrative functions
- Authentication & Authorization levels
- Monitoring & Audit Tracking
- External Interfaces
- Certification Requirements
- Analytics & KPIs Requirements
- Historical Data
- Legal or Regulatory Requirements

4.1.2 Middleware development - First version

The basic development activities will be:

- Base connector implementation

- Interfaces & APIs development
- FIWARE context broker NGSI-LD integration
- Mapping data models into NGSI-LD – First version
- Integration & homogenization middleware sub layer implementation – first version
- User interface base version

The early prototype of the edge-level middleware will be responsible for the basic exchange of information between assets and components and this process will be the main focus of the 1st integration cycle. The prototype will extend the IDSA FIWARE True connector based on the initial functional and non-functional requirements but also the assessment of DEMOs common use cases for data exchange.

4.1.3 Integration & testing 1st cycle

Scope of this phase is to create the OneNet deployment guide which will be part of the new version of the Integration Plan after the first middleware prototype will have been formed. This early prototype will be lab tested to ensure that the OneNet middleware is compliant with the functional requirements according to first phase use cases.

Table 4.2: Phase A: Integration Planning

Phase A: Integration Planning		
1	OneNet Deployment Guide [Integration Guide update]	31/07/22
2	Initial Lab Testing	30/09/22
3	Ensure functional requirements compliance	30/09/22

4.2 Phase B: Second Development Phase (OneNet Reference Platform, Intermediate version)

Table 4.3: Phase B: Deliverables & Milestones

Deliverables & Milestones		
MS16	OneNet Reference Platform, Intermediate Version	31/01/23
D6.1	Report on decentralized edge-level middleware for scalable platform agnostic data management and exchange	31/01/23

The second development phase is split in six distinct stages:

4.2.1 General Functional Requirements Definition - Final version

During this stage the OneNet framework functional requirements will be evolved extending the early prototype scope and analysing all aspects of OneNet. The new set of requirements will focus on the following:

- Definition of Demo specific UCs

- Functional requirements middleware evolution (After the initial data exchange how the middleware will need to evolve taking into consideration all Demo specific Use Cases).
- Workbench & Dashboard final functional requirements
- Integration & homogenization middleware sub layer final functional requirements
- Legal, Regulatory, Privacy and Cyber security Management & Compliance Tools final requirements
- Concluding Contingency plan & Risk management methodology to ensure timely and accurate preventive and corrective process is in place.
- Concluding DPIA: The Integration process relies on the DPIA survey results in order to verify that technologies, workflows and dataflows used are in compliance with GDPR.

4.2.2 Middleware development - Intermediate version

The basic development activities will be:

- Connector implementation & finalization
- Interfaces & APIs development
- Mapping Data Models into NGSI-LD - Final version
- User Interface finalization
- Authentication & Authorization base version
- Integration and Homogenization middleware sub-layer implementation - Final version
- Data quality & harmonization services development

The new edge-level middleware prototype will contain the following components/modules:

1. Semantic Annotation
2. Data Access Policies
3. Data Quality
4. Identity Management
5. Data Catalogue
6. Logging
7. Context Broker (Resource Management)
8. Configuration/Routing
9. Rest API

4.2.3 Workbench & Dashboard development

The basic development activities will be:

- Orchestration workbench implementation
- Monitoring and Analytics Dashboard implementation
- Integration with OneNet framework

4.2.4 Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance first version

The basic development activities will be:

- Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance first version implementation
- Integration with middleware
- Ensure NFR compliance

4.2.5 Ensure reference architecture compliance (first evaluation)

The main objective of this activity will be to ensure the quality and preparedness of all big data tools and separate components to be incorporated in the OneNet Interoperable Network of Platforms hence, to be fully compliant to the Reference Architecture.

4.2.6 Integration & testing 2nd cycle

Scope of this phase is to create the first prototype OneNet platform by integrating all separate components thus creating the OneNet framework. The OneNet platform will be tested both in a lab and field environment and immediately we will perform the initial tests to verify that the prototype is compliant with phase B functional requirements and demo use cases can be performed successfully.

Table 4.4: Phase B: Integration Planning

Phase B: Integration Planning		
1	Testing individual components	28/02/23
2	Testing APIs & Interfaces	28/02/23
3	Integration of all components	28/02/23
4	Integrated platform lab testing	28/02/23
5	Integrated platform field testing	30/03/23
6	Intermediate OneNet prototype demo testing	30/03/23

4.3 Phase C: Final Development Phase (OneNet Reference Platform, Final version)

Table 4.5: Phase C: Deliverables & Milestones

Deliverables & Milestones		
MS13	Updated Report on Cross stakeholder decentralized Data Governance for System Data Exchange	30/03/23

D6.2	Report on Cross stakeholder decentralized Data Governance for System Data Exchange	30/03/23
D6.3	Report on Extended Interoperability and Management with FIWARE	30/03/23
D6.4	Report on AI, Big Data, IoT Orchestration Workbench	30/03/23
MS14	Extended Interoperability Management with FIWARE Final Version	31/05/23
MS15	AI, Big Data, IoT Orchestration Workbench Final Version	31/05/23
MS17	OneNet Reference Platform, Final Version	31/07/23
MS18	Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance Final Version	31/07/23
MS19	Tools for evolution management and Compliance to Reference Architecture	31/07/23
D6.6	Report on Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance	31/07/23
D6.8	Report on OneNet Reference Platform and Components Final Release	31/07/23

The final development phase is split in seven distinct stages:

4.3.1 Functional & Non-Functional Requirements finalization

During this stage the OneNet framework functional and non-functional requirements will be finalized extending the previous prototype scope and taking into consideration issues and possible changes after the 1st integration and testing cycle.

4.3.2 Middleware development - Final version

During this stage the final version of the middleware will be developed based on i) issues identified during 2nd integration and testing cycle, ii) possible new changes dictated by the finalization of functional requirements.

4.3.3 Workbench & Dashboard development – Final version

During this stage the final version of the Workbench & Dashboard will be developed based on i) issues identified during 2nd integration and testing cycle, ii) possible new changes dictated by the finalization of functional requirements.

4.3.4 Tools for Legal, Regulatory, Privacy and Cybersecurity Compliance - Final version

During this stage the final version of the Cybersecurity tools will be developed based on i) issues identified during 2nd integration and testing cycle, ii) possible new changes dictated by the finalization of functional requirements.

4.3.5 Ensure reference architecture compliance (second evaluation)

After the first evaluation regarding reference architecture compliance a new evaluation will take place which will guarantee that the developed components and tools are not only compliant with the RA but also are interoperable against all standards used within the project. Moreover, the tools will be assessed regarding defined quality criteria to guarantee the given TR level and to provide a quality label. Feedback loops will be established with the demonstrators, so that the testing of the tools in real operation will reveal issues and problems that need to be fixed or improved.

4.3.6 Final Assessment

The final assessment of the OneNet Framework will take place.

4.3.7 Integration & testing 3d cycle

Scope of this phase is to extend the first prototype OneNet platform to the final version. The OneNet platform will be tested both in a lab and field environment and immediately after every demo will perform the initial tests to verify that the prototype is compliant with phase C functional requirements and demos will be able to perform their use cases successfully.

- Testing individual components
- Testing APIs & Interfaces
- Integration of all components
- Integrated platform lab testing
- Integrated platform field testing
- Final OneNet prototype testing
- OneNet Validation (in conjunction with C.6)
- WP7 Demos implementation
- WP8 Demos implementation
- WP9 Demos implementation
- WP10 Demos implementation
- OneNet integrated platform validation

Table 4.6: Phase C: Integration Planning

Phase C: Integration Planning		
1	Testing individual components	30/06/23
2	Testing APIs & Interfaces	30/06/23
3	Integration of all components	30/06/23
4	Integrated platform lab testing	31/07/23
5	Integrated platform field testing	31/07/23

6	Final OneNet prototype demo testing	31/07/23
7	OneNet Validation (in conjunction with C.6)	31/07/23
8	WP7 Demos implementation	30/09/23
9	WP8 Demos implementation	30/09/23
10	WP9 Demos implementation	30/09/23
11	WP10 Demos implementation	30/09/23
12	OneNet integrated platform validation	30/09/23

5 Contingency Plan

The Contingency Plan of OneNet spans over the following aspects:

- OneNet design
- OneNet development
- OneNet deployment & testing

The Contingency Plan addresses the measures of mitigating certain risks that might arise during any phase of the development and integration of the OneNet system. Such mitigation measures have been embedded in the Integration plan and the system as it is being designed, developed, tested, deployed, and documented throughout the project phases.

Kind of Risk	Description	Impact	Mitigation Measures
<u>Overall Development Management</u>	OneNet as a system is pivotal for the overall OneNet project, in fact it is the system that will prove and promote the inter-platform communication and coordination of flexibility services.	High	Constant communication with the stakeholders through regular and irregular teleconferences, Emails and documents exchange.
<u>Timing</u>	Timing refers to alignment of actions. The complexity the project with 10 demos and ca. 70 different partners participating in the respective WPs (WP6 to WP11) requires that all participants follow the anticipated schedules or inform as early as possible on possible deviations	High	Constant communication through the various communication channels established within the project. Timely identification of deviations from the anticipated integration plan. Revision of the integration plan's time schedule as appropriate to minimize the effect of the deviations.
<u>Stakeholders' engagement</u>	Demos are being considered as the end-beneficiaries of OneNet. As such their constant engagement in the various phases, in certain cases of external entities that are not part of the consortium (e.g. Flexibility Service Providers that have been integrated to test	High	Besides communication, distribution of evolving documentation, live presentations of the system from the early development phase, establishment of structured communication channels (i.e. GitHub).

	the Connector), beginning from Design up to testing, is important.		
Design			
Scope (SUCs) accuracy & variations	The System Use Cases comprise the backbone for OneNet's design. The accuracy and efficiency of the SUCs will result also to an efficient and accurate OneNet Middleware. However, as the scope of OneNet is new to the involved stakeholders/beneficiaries, it is possible that in the due course of detailing and development the UCs might need to be revised or adapted. A risk is eminent if this revision process continues beyond the timeframe of the testing and if the development is confronted with OneNet adaptations/integration that effect essential parts of the system design (data model, APIs, processes) and not only limited operational aspects.	Very high	Early involvement of stakeholders' experts during SUCs design. Detailed documentation of SUCs, usage of potential good practices or standards
Differences among Demos	OneNet envisages to be a unique system and not a customized system for each demo. As such it is expected that certain design characteristics (especially SUCs) shall be common among the demos. Particularities on similar characteristics among the Demos increase design and development complexity.	High	Communicate development to the demos through: <ul style="list-style-type: none"> - Updated documentation relative to D5.3 (Cross Platform Services) [3] and D5.6 (Business Objects) [6] - Connector prototyping - Online UI mock-up - Presentation and Discussion of the system from the Users' perspective

Users' acceptance	Users must accept the system. This means that the system shall cover the functions that it has been designed for and shall also be easy to understand and to use. The risk is that users may not easily familiarize with the system or the system does not correspond to the user's (reasonable) expectations or demands.	High	Communicate development to the users through: <ul style="list-style-type: none"> - Updated documentation relative to D5.3 (Cross Platform Services) [3] and D5.6 (Business Objects) [6] - UI (Workbench/Dashboard) prototyping - Presentation and Discussion of the system from the Users' perspective
<u>Development</u>			
Timing	Timely provision of development results and intermediate results. Possible schedule overruns (especially for critical identified parts) in development might affect the overall project schedule but also the quality of the system.	High	Follow and monitor the development time-plan, make intermediate results available to other development parties as early as possible. Use the established channels for seamless communication, respond asap on comments from other participants.
Specifications' fulfilment	All functional and non-functional requirements must be met during implementation.	High	A Requirements Traceability Matrix (RTM) shall be developed for every release. Respective Tests (in accordance with the Testing Guidelines) shall confirm the fulfilment of the requirements.
Tools & Technology Stack	Use tools and technologies that do not impede deployment beneficiaries and users from accessing the system or require skills that are beyond the usual business IT infrastructure for the electricity domain	High	Select standard, and if possible open-source, tools and technologies. Inform Demos as early as possible on the proposed technology stack.

<u>Integration</u>			
Local Services & Local Components	If the integration with local services and components is not seamless, i.e. to exchange correct information in the correct time, then the system integration will practically fail.	High	Develop the necessary middleware that will integrate local components and local services with the core system, upon comprehensive API definition, implementation and testing.
APIs Documentation	Incomplete API Documentation might impede the integration of local actors and systems	High	Provide detailed API documentation.
APIs accessibility	API accessibility shall be ensured for the actors that need to address it	High	Documentation and extensive testing shall ensure that API accessibility is ensured for the actors that need it.
<u>Deployment</u>			
Hosting Infrastructure	The hosting Infrastructure shall be adequate for the technologies and the payload that the system requires.	High	Select a technology Stack that provides for a usual hosting environment that any contemporary IT department of a medium-large organization has.
Network Security	Security of the hosting infrastructure is of ultimate importance as OneNet system is being exposed to the Internet.	High	Follow the guidelines of D5.8 [7] and D6.6.
Logging & Debugging	A new IT system might show programming errors (bugs) or malfunctions that might arise from the extensive testing of the system with users and its integration with external systems, under several edge conditions and parameterizations.	High	Embed an extensive (and expandable) logging reporting system into OneNet system. Implement a debugging strategy for each demo during the integration, testing and piloting phases.

6 DPIA Survey Methodology and Plan

A Data Protection Impact Assessment (DPIA) must be conducted prior to any integration process for all different data processing activities and software components, in accordance with the current legal framework in the EU (General Data Protection Regulation) and the guidelines applicable for the project, common requirements set in the “*Horizon 2020 - Work Programme 2018-2020 Secure, clean and efficient energy*” page 10 [9]:

Projects dealing with data handling and management should comply with the provisions of the General Data Protection Regulation¹ and industry standards, e.g. the **Data Protection Impact Assessment Template**². A high level of cyber security should be guaranteed in compliance with relevant EU security legislation³ and with due regard of best available techniques for ensuring the highest level of protection⁴

and chapter 3 of D5.8 [9] of OneNet project where data privacy concerns are addressed through a data protection-by-design and ..-by-default approach followed by the OneNet data processing activities/procedures to ensure that no legal or ethical issues arise during the project execution.

Risk management measures are in place to address issues identified for data flows that process *personal* information. Such risk management (mitigation) procedures are documented in the DPIA conducted (see attachment) and are briefly presented in the following paragraphs.

6.1 Implementing DPIAs risk mitigation

6.1.1 User Registration and Consents

- User Registration is not possible if user does not provide a signed informed consent form as considered necessary in every pilot.
- Consent form thoroughly analyses the reasons for obtaining the personal data and how they are going to be processed/utilized.
- Consent form explicitly states how a user can withdraw hers/his consent.
- Users can access their personal data both through:
 - (a) the OneNet Workbench/Dashboard
 - (b) commonly used electronic format in case the request is made electronically.

¹ Regulation (EU) 2016/679

² Supported by the Commission Recommendation 2014/724/EU

³ Directive (EU) 2016/1148

⁴ Best Available Techniques for data protection and security regarding the ten minimum functional requirements for smart metering which were proposed in the Commission Recommendation 2012/148/EU.

- Users can update personal data through the OneNet Workbench/Dashboard.
- Consent form explicitly states that personal data are collected for research purposes.
- Include the data retention policy in the privacy notice.

6.1.2 Technical measures

- ISO27001:2013 certification of the infrastructure that stores and processes user data
- Store only the necessary user information
- End-to-end encryption for data in transit. Data in transit is protected against active (e.g. replays, traffic injection) and passive attacks (e.g. eavesdropping), thus ensuring data integrity.
- Database encryption for data at rest (example: TDE – Transparent Data Encryption)
- Pseudo anonymization (IT-System that contains the relations is maintained securely off-line)
- Extensive logging

6.1.3 Organizational Measures

The following paragraphs give general guidelines and must be verified by the respective legal departments of the involved partners in order to comply with local legislation as necessary while piloting.

Organizational Measures refer mainly to implementation of the data protection policies that are described in D5.8 [7] and in first line with compliance of the GDPR. Data processes in connection with personal data shall be analysed in the DPIA and among the measures, as provided in the GDPR are the conclusion of Joint Controllers Agreement(s), Data Processing Agreement(s) and Non-Disclosure Agreement(s), if necessary on each case.

Dealing with test-data or anonymized data might not necessarily require strict alignment towards conclusion of the aforementioned agreements. Dealing with real data however implies that the respective agreements have to be concluded.

- **Joint-Controllers Agreements (“Arrangements”)**

In accordance with the GDPR ‘controller’ means the ... legal person... which, alone or jointly with others, determines the purposes and means of the processing of personal data. In case of OneNet, Controller might be each one of the Platform Operators that will assume responsibility of operating a local application (instance) of the OneNet Connector.

- **Data Processing Agreements (“Arrangements”)**

In accordance with GDPR ‘processor’ means a natural or legal person, ... which processes personal data on behalf of the controller; this means that the OneNet connector deployment and the entities responsible for hosting and maintain it shall consider whether they assume

the role of a data processor. The same might apply beyond the OneNet system to the information that will be exchanged between platforms (and which might be agnostic to the OneNet Middleware): the respective data consumer/producer entities (i.e. platforms) shall assess whether they assume the role of a data processor.

- **Non-Disclosure Agreements**

NDA's might deem necessary if third parties are involved in certain activities that do grant them access to data and especially sensitive data, but not personal data. Please note that this is not necessary among project partners, as the Consortium's Agreement covers extensively with the provisions in Chapter 10 the obligations of non-disclosing information.

6.2 DPIA sample for OneNet

The following Table 6.1 presents the structure of a DPIA that needs to identify the data flows, the effected data subjects and the data elements processed within OneNet.

Table 6.1: DPIA Structure

Mapping of Personal Data Flows	Name of data flow	Please indicate the name of data flow
	Short description of data flow	Provide short description of data flow
	Department (Business Unit)	Please indicate the name of the department that is processing the personal data
	Data Storage	How the data is stored? e.g. Data Location, Database, Server
	Format	Please provide the format of the information e.g. paper, electronic, structured, unstructured
	Employee(s)	Indicate the name of the employees that are acting as processors in the relevant department
	Document Coming From	e.g. provided by staff
	Data Processor (if applicable)	If applicable, please indicate the name and contact details of the data processor, if there is a data processing agreement in place, the location of the server of the data processor and the details of the subprocessors (if any).
Article 30 documentation (mandatory)	Data controller and DPO details	For each data flow please add: - Name and contact details of data controller - Name and contact details of Data Protection Officer (DPO)
	Purpose of the processing	Please describe the purpose of collecting, processing and storing the personal data in this data flow <i>If several purposes can be identified please write all of them e.g. recruitment purposes, employee administration, financial services administration etc.</i>
	Categories of data subjects	Please specify the categories of data subjects (i.e. whose personal data is processed in this flow) e.g. Employees, Shareholders, Job applicants, Former Employees, Consultants, Clients, Suppliers
	Data elements	For each category of data subjects, please specify the categories of personal data covered by the data flow. Ordinary data such as: name, address, e-mail, phone number, gender, age, picture, IP address, initials, personality tests etc. e.g. Re Employees: name, address, gender, age, information in CV, references, etc.
	Special categories of data, criminal convictions and offences and/or national identification numbers	For each category of data subjects, please specify whether this flow entails the processing of: (i) any special categories of data (i.e. personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership , and the processing of genetic data , biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person's sex life or sexual orientation) (ii) data regarding criminal convictions and offences and/or (iii) the processing of national identification numbers .
	Categories of recipients to whom the personal data have been or will be disclosed to	Please describe the categories of recipients whom you share the data with (this could be both data processors and independent data controllers) e.g.: Officers/directors, HR manager, HR administration staff, IT administrators, IT Service Providers, authorities, companies, partners etc.
	Country of recipient(s) if outside the EU/EEA	Please identify all recipients of personal data in this flow who are located <u>outside EU/EEA</u> , and provide name, address, company registration number (if relevant), contact details and country of such recipient. <i>Please remember this can be both data processors, sub-data processors and/or independent data controllers. For example for cloud providers; please consider where the servers are located.</i>
	Data category assessment (private, sensitive, open)	Indicate the Category of Data to be assessed
	LEGAL ASSESSMENT REQUIRED Legal basis for the transfer of personal data outside the EU/EEA	If the data is transferred outside the EU/EEA a legal basis for transfer of personal data is required (Art. 45 - 49). Please identify legal basis for each transfer outside the EU/EEA. <i>If relevant, add more legal basis. If applicable (namely if Second subparagraph of Article 49 (1) is relied upon for the transfer, please describe documentation of suitable safeguards.</i>
	LEGAL ASSESSMENT REQUIRED Retention period	For each category of data, please describe the retention period. If the retention period is defined in EU or local legislation, please provide reference to the relevant legislation e.g.: Health data kept for 30 years Contact data kept for 5 years after xx Data on job applicants who did not get the job; kept for 6 months.
	Technical and Organisational measures	Please describe the technical and organisational security measures which have been applied in this data flow <i>EXAMPLE: Encryption or pseudonymisation of data, firewalls, the use of specific IT security standards, strong passwords, limitations in access, locks etc.</i>
Art. 6 General Requirement	LEGAL ASSESSMENT REQUIRED Legal basis for processing	For <u>each data flow</u> identify what is the legal basis for lawfully processing the personal data: (a) consent; (b) processing is necessary for the performance of a contract to which the data subject is party or in order to take steps at the request of the data subject prior to entering into a contract; (c) processing is necessary for compliance with a legal obligation to which the controller is subject; (d) processing is necessary in order to protect the vital interests of the data subject or of another natural person; (e) processing is necessary for the performance of a task carried out in the public interest or in the exercise of official authority vested in the controller; (f) processing is necessary for the purposes of the legitimate interests pursued by the controller or by a third party

7 The OneNet Connector – Technical Characteristics

The OneNet connector, as described in D5.2 [2], is the core component capable of enabling decentralized data exchanges in the OneNet ecosystem. The OneNet connector is a deployment instance of the OneNet Decentralized Middleware and once deployed and integrated within the platforms of each OneNet participant, it allows to create a trusted pan-European data space defined as OneNet Network of Platforms.

The development of the first version of the OneNet Connector, guided in task T6.1, followed all the functional and non-functional requirements collected starting from the different use cases described in D5.1 [1] and D5.4 [4].

This chapter contains a general technical overview of the OneNet Connector, which will be further detailed in D6.1 (expected at M28). In addition, are reported the main interactions and scenarios envisaged for this first release and which will be tested and validated, according to the integration plan, in the environments of the various demos.

7.1 OneNet Connector Architecture

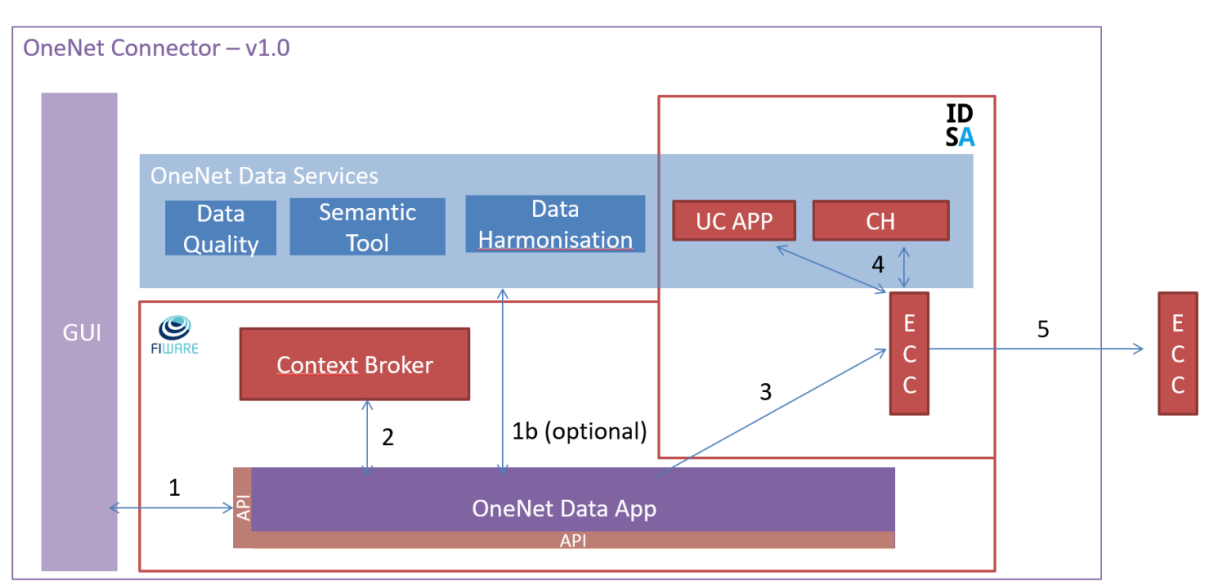


Figure 8: OneNet Connector Architecture - v1

As shown in the Figure 8, which shows the overall OneNet connector architecture, the Connector is composed of several components:

- GUI, is the graphic interface that allows to configure the OneNet Connector and use the main features. Each OneNet participant can access their own connector's GUI once installed. The first version of the GUI includes:

- o Registration as data provider / consumer
- o Creation of new data offerings

- o Creation of data entities
- o Registration / consumption of data offerings and data entities
 - OneNet Data App, provides integration APIs for both the GUI and external platforms. It is based on the NGSI standard for the implementation of APIs and for data exchange and in this first version it offers three main services:
 - o Entity creation
 - o Registration / Subscription
 - o Data retrieving
 - Context Broker, is the FIWARE component that manages the overall data exchange in the standard NGSI-LD format
 - IDS based components, a series of components and tools, extended from the open-source TRUE Connector [12], that enable the IDS processes for the data exchange
 - o Execution Core Container (ECC), has the task of implementing IDS-based data exchange processes (authentication, metadata exchange, access control, logging, etc. ...)
 - o Clearing House, logging system of all data exchanges (not available in the first version but being integrated)
 - o Usage Control App, module for verifying access and use of data (not available in the first version, but being integrated)
 - OneNet Data services, additional data services provided by the OneNet Connector (additional information is provided in D5.3 [3]). They are not included in this first version.

7.2 OneNet Connector – v1.0

7.2.1 OneNet GUI

The GUI provided enables the OneNet users:

- To define certain configuration settings for the user and its OneNet connector
- To streamline the Service Management (*cross-platform-services provision and subscription*) and the data exchange (*provision and consumption*).
- To use the OneNet connector's APIs

A comprehensive presentation of the GUI in the form of a manual is presented in the Annex. A. The most recent version of the GUI users' manual will be accessible online through a dedicated menu item on the GUI.

7.2.2 OneNet Connector Data App

The main goal of the OneNet data app is to be the entry point of the OneNet Connector. In fact, it provides access to all the main features of the OneNet Connector, including those for data exchange, through standard interfaces based on REST API.

From a technical point of view, the OneNet Data App is developed using a Java Architectural Stack, based on Spring Boot [10] framework and a NoSql database, MongoDB [11].

All the services made available by the OneNet Data App can be used both from the GUI of the connector and directly from the external platforms that need to be integrated with the OneNet Connector.

The first version of the OneNet Connector Data App implements three main interactions, provided as standard REST API and available using the GUI (see Ch.7.2.1 and the Annex) or the API interfaces directly (see documentation at Ch.7.2.4)

These three interactions are:

- **Create Entity**, a data provider is able to create new entities within its own environment and makes them available to all the other OneNet Participants
- **Registration**, a data consumer is able to register to specific data entities for receiving data in automatic or manual way, after the acceptance of the data provider. The data consumer can register itself to many data providers and many data entities.
- **Get Entity**, the data consumer, after the registration, is able to retrieve specific data entity from the data provider

For each interaction a high-level sequence diagram is reported below.

Create Entity

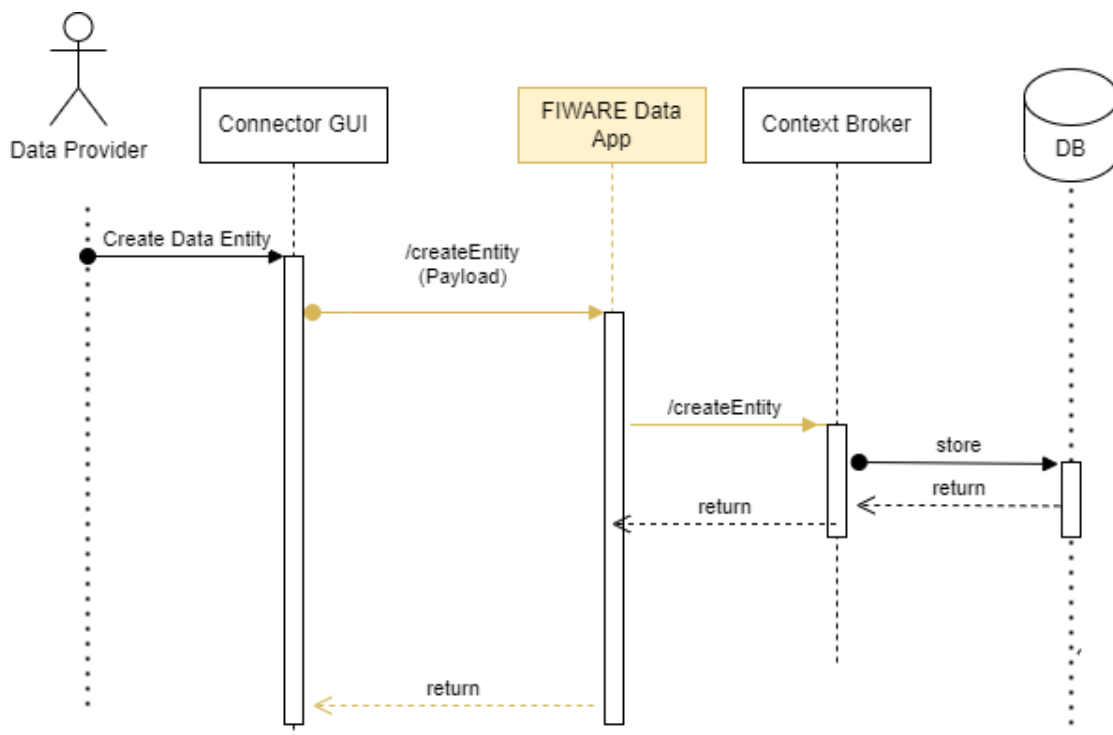


Figure 9: Create Entity - Sequence Diagram

Registration

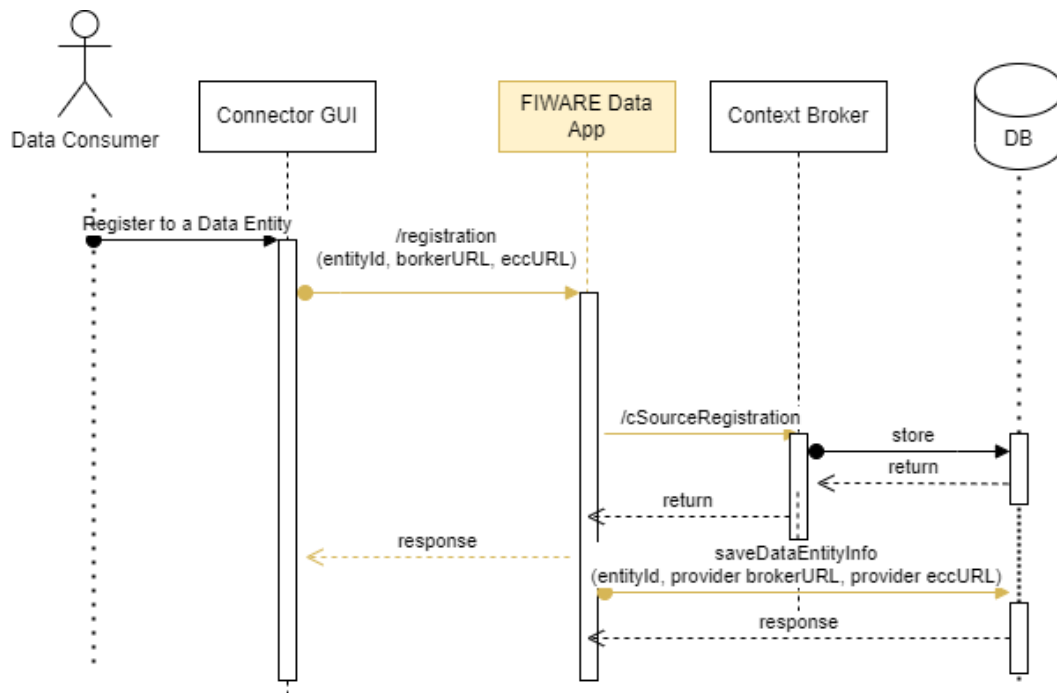


Figure 10: Registration - Sequence Diagram

Get Entity

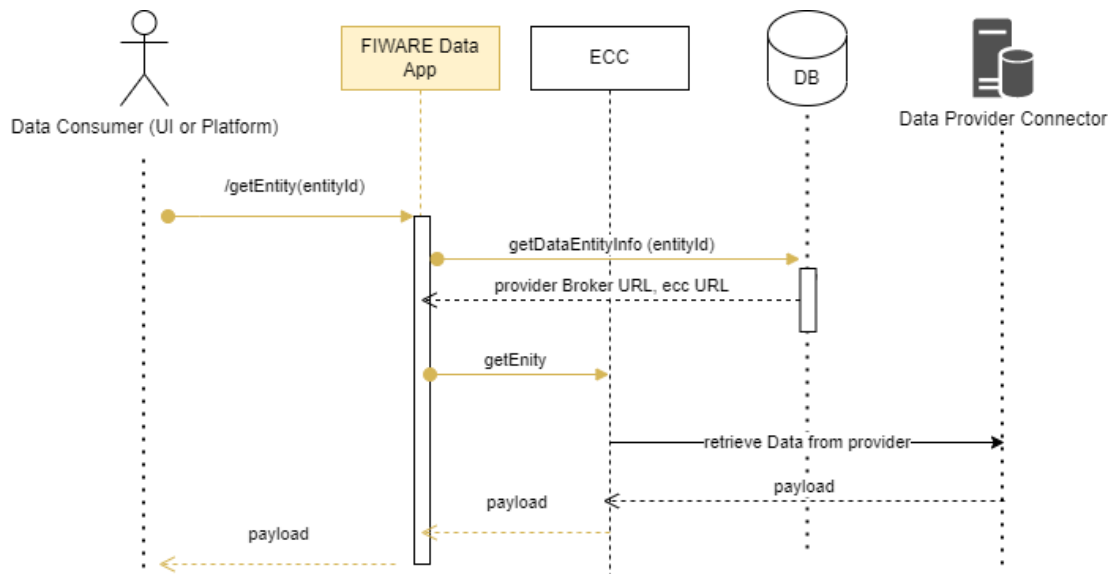


Figure 11: Get Entity - Sequence Diagram

7.2.3 FIWARE and IDS based Components

The core implementation for handling the data exchange in the OneNet Connector leverages on the integration of FIWARE components (FIWARE Orion Context Broker, using NGSI-API) and the IDS Reference Model (IDS based connector).

FIWARE Orion Context Broker, is one of the core components of the FIWARE framework and implements the Next Generation Service Interface (NGSI) standard to implement data exchange mechanisms using standardised data models and interfaces. In FIWARE framework, Orion Context Broker is a key service to ease the development and provisioning of smart and innovative applications that require context information and data stream management, processing and exploitation.

In the OneNet Connector implementation, the FIWARE Orion Context Broker-LD [13], which supports NGSI-LD and NGSI-v2 API, is in charge of managing all the data exchanges among the OneNet participants in a standardised way.

The broker is integrated with the OneNet Data App and with the IDS based components using standardised NGSI APIs and supporting standardised data models based as described in D5.4 [4].

As described in D5.2 [2], the OneNet Architecture follows the IDS reference model for the implementation of a pan European data space, leveraging on existing standards and technologies, as well as governance models to facilitate secure and standardized data exchange and data linkage in a trusted business ecosystem.

In the IDS reference model, the IDS Connector is a key component for implementing a European Data Space and the OneNet Connector implements an extended version of the IDS based TRUE (TRUsted Engineering) Connector [12].

In this context the evolution of the TRUE Connector in the OneNet Connector allows to implement a FIWARE-based Data Space, integrating the existing FIWARE Ecosystems in a plug-and-play mode enabling the creation and support of IDS ecosystems.

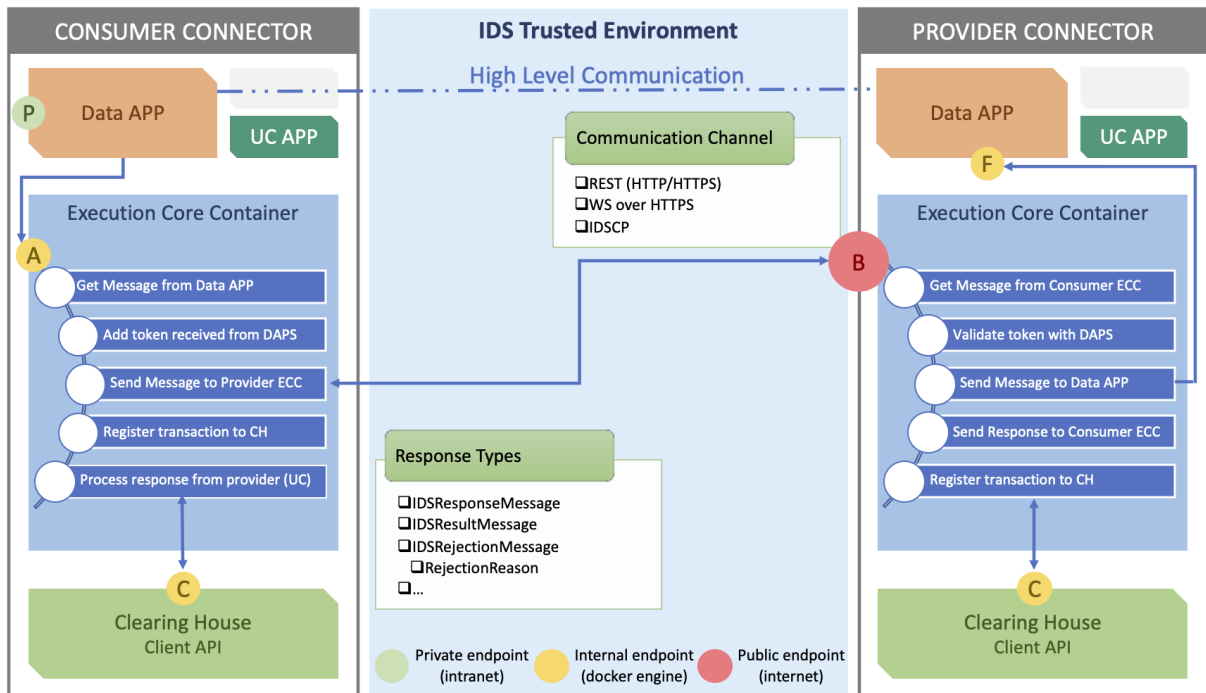


Figure 12: IDS based TRUE Connector [12]

The core of the IDS implementation of the OneNet Connector is the Execution Core container, that is compliant with the latest IDS specifications (IDS Info Model) and can be easily customized to fit a widespread number of scenarios.

It is already integrated with the OneNet Data App and FIWARE Orion Context Broker-LD will be integrated with other existing IDS core services (Identity Providers, Clearing House, Usage Control App) in the next version (expected at January 2023).

7.2.4 REST-APIs

All the features made available in the OneNet Data App are provided through standard REST APIs. The APIs are available both for the Graphical User Interface and for the external platforms, to be used for the integration. The table below report a short description of the APIs exposed so far. A detailed and standardised OpenAPI⁵ documentation will be provided together with the software release.

Table 7.1: OneNet Connector REST APIs

Name	Url	Method	Parameters	Responses
Create Entity	/createentity	POST	In Body (JSON format) payload : String	Success (200) Error (500)

⁵ <https://swagger.io/specification/>
Copyright 2022 OneNet



				Error Message - <i>String</i>
Registration	/registration	POST	In Body (JSON format) entityId: String eccUrl: String brokerUrl: String	Success (200) Success Message: <i>String</i> Error (500) Error Message: <i>String</i>
Get Entity	/getentity	GET	In <u>Request</u> : entityId: String	Success (200) entity Error (500) Error Message - <i>String</i>

7.3 Communication Channels

As the OneNet platform adheres to the IDSA Reference Architecture, it also follows specific principles. One of the most important ones is the “Trustworthy Communication & Security by Design”. Specific software components like Connectors, App Stores, and Brokers can check if the Connector of the connecting party is running a trusted (i.e. certified) software stack. Any communication between (external) Connectors can be encrypted and integrity protected. Each Data Owner and Data Provider must be able to ensure that their data is handled by the Connector of the Data Consumer according to the usage policies specified: otherwise the data will not be sent. To reduce the impact of compromised applications, appropriate technical measures must be applied (e.g. isolating Data Apps from each other and from the Connector). Data Providers and Data Consumers can decide about the level of security to be applied for their respective Connectors by deploying Connectors supporting the selected security profile.

Responsible for supporting Communication is the **Execution Core Container** which provides components for interfacing with Data Services (e.g., Data Router or Data Bus to a Connector).

- A Data Router handles communication with Data Services to be invoked according to predefined configuration parameters. In this respect, it is responsible of how data is sent (and received) to (and from) the Data Bus from (and to) Data Services. Participants have the option to replace the Data Router component by alternative implementations of various vendors. Differences in configuration can be handled by specialized Execution Configurator plug-ins. If a Connector in a limited or embedded platform consists of a single Data Service or a fixed connection configuration (e.g., on a sensor device), the Data Router can be replaced by a hard-

coded software, or the Data Service can be exposed directly. The Data Router invokes relevant components for the enforcement of Usage Policies, e.g. a Policy Enforcement Point (PEP), as configured in the connector or specified in the Usage Policy.

- The Data Bus exchanges data with Data Services and Data Bus components of other Connectors. It may also store data within a Connector. Usually, the Data Bus provides the method to exchange data between Connectors. Like the Data Router, the Data Bus can be replaced by alternative implementations in order to meet the requirements of the operator. The selection of an appropriate Data Bus may depend on various aspects (e.g., costs, level of support, throughput rate, quality of documentation, or availability of accessories).

To ensure confidentiality and authenticity of data transfers, communication between Connectors must be protected. When using the IDS Connector, two layers of security are in place: i. point-to-point encryption (between Connectors), using an encrypted tunnel, and ii. end-to-end authorization (authenticity and authorization based on actual communication endpoints; i.e., Data Apps). Data from one External Connector to another is sent over the Internet or via a virtual private network (VPN), the specification of which is beyond the scope of the IDS Security Architecture. The Security Architecture defines the IDS Communication Protocol (IDSCP), which must be supported by Trusted Connectors, and can be supported by any other Connector as well. The purpose of the IDSCP is to establish confidential, authenticated communication, exchange data between the Data Provider and the Data Consumer, and establish mutual remote attestation (if supported by the Connectors involved).

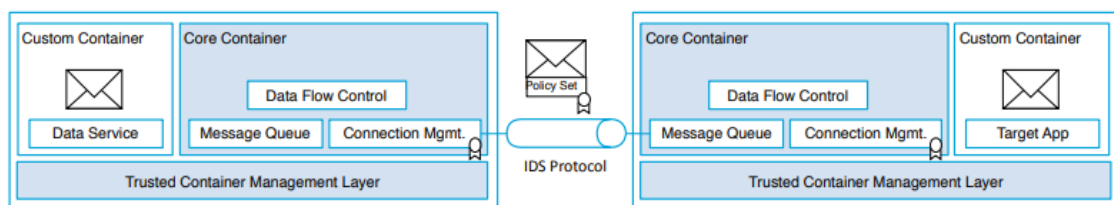


Figure 13: Secure Communication

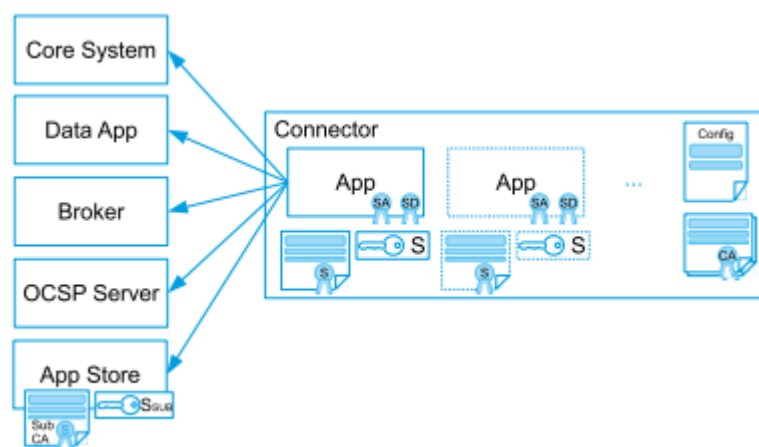


Figure 14: Connector Roles and Manifestations

8 Integration testing activities for OneNet

8.1 Introduction

Software testing is the process of evaluating a software product to find whether the current product meets the required conditions or not and to ensure that the software product is defect-free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements. Software testing is crucial for the early identification of errors in a software product before the specific version gets released to the customer or end-user. The benefits of testing are summarized below:

- Software Quality: An essential requirement for any software system with the intention of high-quality software delivery to the end-users and key stakeholders.
- Cost minimization: Testing a project on time helps save money and time in the long term. The earlier the bugs are caught, the less it costs to fix them, and that can also mean that there are less cycles between the developer and the customer (the pilot demonstrators in our use case).
- Customer satisfaction: The main aim of any product is to give satisfaction to their customers, increase their engagement and improve the user experience.

Traditionally testing is divided into three main categories: Functional, Non-Functional and Maintenance. Though today there exist more than 150 types of testing methods, the more widely used (and the ones that we are going to address in OneNet) are the following:

- Functional Testing: UI Testing, API Testing, Integration Testing
- Non-Functional Testing: Performance Testing
- Maintenance Testing: Regression Testing

In the first development phase we will concentrate around functional testing only.

Functional Testing

Functional testing is a type of black box testing that evaluates the conformance of a system or component with stated functional requirements. Functional testing specifies what the system does. Each functional test verifies if all the functionalities of the application are working as expected and ready for release. Since it is black box testing, the tester will not know the internal structure of the app or the source code. Development team will create functional test cases with respect to user/business requirements and testers specify the functionality requirements based on the user perspective. Functional testing can be carried out by manual test teams or can be automated.

Under the OneNet scope, functional testing will be based on the OneNet Functional Requirements derived from task T6.1. These high-level requirements represent the technical functionalities that the OneNet participants envisage to get from the OneNet System components and can be further divided into smaller sub-requirements. We should note here that each sub-requirement refers to a specific OneNet component e.g. Connector: User Interface, as analysed in T6.1.

Three different types of functional testing will be addressed in OneNet: UI Testing (frontend), API Testing (backend), Integration Testing (frontend-backend). This approach follows the typical web application architecture where three separate architectural layers exist: the presentation (or user interface) layer, the business layer (where all the business logic is handled) and the database layer for data modelling and manipulation.

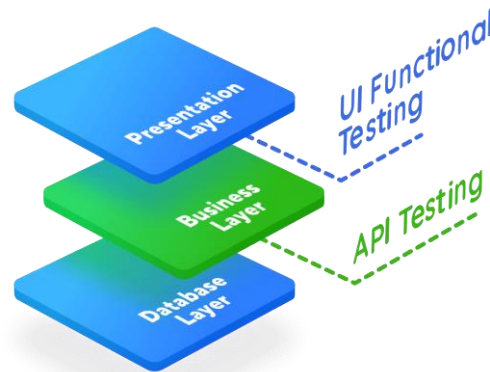


Figure 15: The three layers of the OneNet web application

UI Testing, performed at the presentation layer, is a technique for testing the features of any software that a user will interact with. This usually involves testing the visual components to ensure that they are meeting the outlined requirements - both in terms of functionality and performance. UI testing is centred around two main things. First, checking how the application handles user actions carried out using the keyboard, mouse, and other input devices. Second, checking whether visual elements are displayed and working correctly. Some examples (not an exhaustive list) of what you can test with UI testing are summarized below:

- Menu items – Ensure the application only displays valid menu items that are available at a particular state.
- Navigational elements – Verify all navigational buttons on a selected page are working correctly, and that they redirect users to the right page or screen.
- Navigational elements – Verify all navigational buttons on the page are working correctly, and that they redirect users to the right page or screen.
- Data type errors – Ensure only valid data can be entered for specific data types such as currency and dates.
- Field widths – If a certain text box permits a specified number of characters, then make it clear on the user interface that the data entered shouldn't exceed the character limit.

API Testing, performed at the business layer, involves all the logic and all transactions between the user interface and database layers. It is a software testing practice that tests the APIs directly — from their functionality, reliability, performance, to security. Part of integration testing, API testing effectively validates the logic of the build architecture within a short amount of time. In API Testing, instead of using standard user inputs (keyboard)

and outputs, you use software to send calls to the API, get output, and note down the system's response. API tests are very different from GUI Tests and won't concentrate on the look and feel of an application. They mainly concentrate on the business logic layer of the software architecture. Some examples (not an exhaustive list) are presented below:

- **Return value based on input condition:** it is relatively easy to test, as input can be defined, and results can be validated.
- **Trigger some other API/event/interrupt:** If an output of an API triggers some event or interrupt, then those events and interrupt listeners should be tracked.
- **Modify certain resources:** If an API call modifies some resources, then it should be validated by accessing respective resources.

Finally, Integration Testing, is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. For the purpose of OneNet we will use integration testing between the frontend and backend components in order to ensure that the list of functional requirements of 6.1 are met end-to-end.

Because of the architecture of OneNet the integration testing can be described with the sandwich testing strategy (Figure 16). Top level modules (module 1 is equivalent to OneNet Connector Dashboard) are tested with lower-level modules (modules 2,3 are equivalent to the components of OneNet Middleware whereas modules 4,5,6 are equivalent to the OneNet Connectors applied to demo legacy platforms_ at the same time.

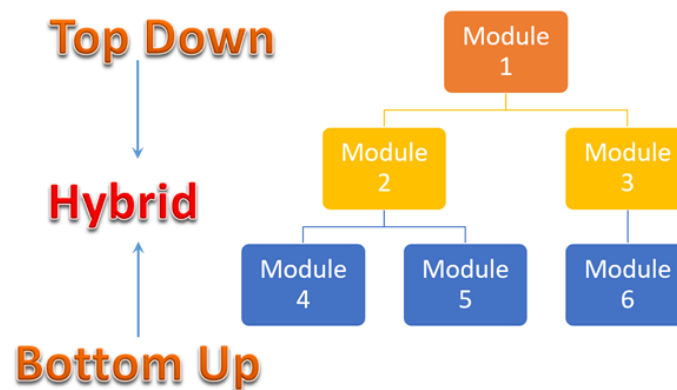


Figure 16: Sandwich testing strategy for OneNet ecosystem

The Integration test process, which is further elaborated in the following sections , is summarized in the following steps:

- 1) Prepare the Integration Test Plan.
- 2) Design the Test Scenarios/Cases.
- 3) Choose appropriate framework and write the tests.
- 4) Execute the test Cases followed by reporting the defects if any.
- 5) Track & re-test the defects.

- 6) Steps 3 and 4 are repeated until the completion of Integration is successful.
- 7) Choose appropriate automation framework in order to automate the tests (optional).

8.2 Test Plan and Example Test Case Template

A test plan is a document describing the scope, approach, resources and schedule of intended testing activities as per ISTQB definition. The test plan serves as a blueprint to conduct software testing activities as a defined process. The seven tests for a successful test plan as per IEEE 829 are shown in Figure 17. We will analyse each step and adapt it specifically to the OneNet purposes and needs.

Step 1) Analyze the product

The product analysis usually answers questions of type “who and how many are the product users”, “what is the product used for”, “how will the product work” etc. In our case the OneNet participants are interested in data exchange functionalities through the successful connections of their legacy systems to the OneNet middleware. Monitoring of traffic, user activities, platform activities and basic analytics are also of high interest. Participants can mainly be divided into data owners and data consumers. According to the measurable objectives of the project, by the end of the project the number of active users on the customer side should reach the 200. To better estimate the anticipated traffic in the OneNet web application, for each requirement finalized in 6.1 a direct mapping should be made with the list of cross-platform services available in D5.3 [3] and the respective methods (Get/Post/Put/Delete) that it will provide. For example:

Final list of requirements = 25

Services per requirement = 5 (average estimation)

Methods = 4 (Get/Post/Put/Delete)

Users = 200 (Assuming target is reached)

The maximum anticipated traffic per day assuming all users are logged in each day and they perform at least one request from each category: traffic = $20 \times 5 \times 4 \times 200 = 100.000$ requests per day.

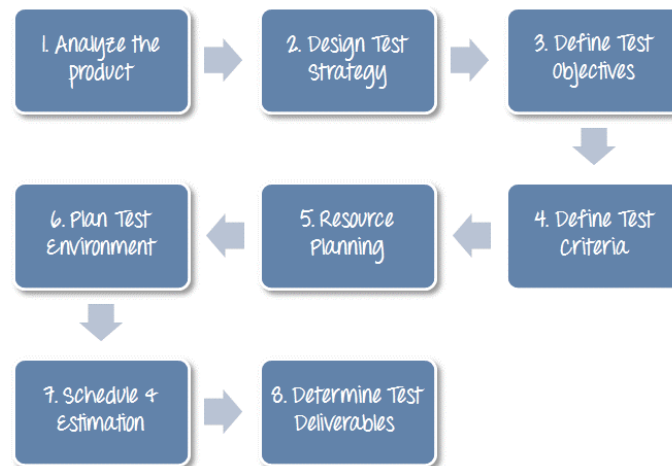


Figure 17: Steps to create a test plan

Step 2) Develop Test Strategy

In this critical phase the testing scope should be defined, an appropriate choice of testing type must be made, documentation of any identified risks and issues must be clarified and creation of the test logistics must be completed.

In the first development phase of OneNet, we will only concentrate on Functional Testing as stated before, in other words functional testing is in-scope. Performance testing for instance is out of scope for this phase. The testing types that we have chosen are UI testing, API testing and Integration testing.

As potential risks we can identify the following ones in OneNet:

- Project schedule is too tight, and it might be hard to complete all the levels of testing for all the components.
- API Testing is part of the corresponding development component, so the owner of the component development is also responsible for testing it. It might be the fact that there is a lack of required skills for testing.

Finally, for the design of the test strategy step, we must define who will be the tester and when will the tests occur. Each technical development partner (ED, ENG, UBE/UBI) must appoint a team member who will take charge for the test execution and who will be the main point of contact for the testing activities. This person is responsible to choose the tech stack/testing framework of his preference, prepare the testing environment and start the recurring testing activities. It would be good for the project to have clearly defined development and test cycles as part of a smooth agile process (e.g., 3 weeks development, 1 week testing).

The third step of the test plan creation is the definition of the test objective, which is the overall goal and achievement of the test execution. At the first development phase we cannot clearly define that since the technical requirements are still under construction.

The fourth step of the test plan creation is the definition of test criteria. The suspension criteria are the most critical criteria, and the active test cycle is suspended until these criteria are completely resolved. An example

would be that at a certain testing cycle there is a critical amount of testcases failing like 30%. So that means that the testing cycle is suspended until the development team fixes all the bug related to the massive test failure. Then the testing cycle resumes. The exit criteria specify a successful completion of a test phase. The exit criteria are the targeted results of the test cycle and are necessary before proceeding to the next phase of development. In our case we will define a **95%** of all critical test cases pass rate.

The fifth step of the plan is the clear definition of all types of resources required to complete testing tasks, both human as well as system resources. Each technical partner in OneNet is free to define both for his own team.

Planning the test environment is the sixth step of the plan. A testing environment is a setup of software and hardware on which the testing team is going to execute test cases. This is again up to the technical partner to decide. In Ubitech all the tests will be executed automatically in our CI/CD environment where a dedicated GitLab server is used.

In the test schedule and estimation phase (7th step), the whole project is broken into smaller tasks and an estimation is added for each task as the example below:

Task	Members	Effort
Create the tests	Test Designer	170 man-hour
Perform the test execution	Tester	70 man-hour
Test Report	Tester	10 man-hour
Test Delivery		10 man-hour
.....	
Total		

In the final step of the test plan, the test deliverables must be clarified. Test Deliverables is a list of all the documents, tools and other components that has to be developed and maintained in support of the testing effort.

Test deliverables that will be provided **before** testing phase for OneNet:

- Test plan document (as part of the Integration Guide)
- Test cases documents

Test deliverables that will be provided **during** the testing phase:

- Test Scripts
- Test tools and framework
- Test Data

- Error logs and execution logs

Test deliverables that will be provided **after** the testing cycles is over.

- Test Results/reports
- Software Defect Report

After concluding all the mandatory steps of the test plan, we are now in the position to present how a test template would look like:

Test Case ID	TC_01	Test Case Description	Verify that a new OneNet participant can be successfully registered in the dashboard.
--------------	-------	-----------------------	---------------------------------------------------------------------------------------

Tester's Name	Dimitra Georgakaki	Date Tested	1-March-2022	Test Case (Pass/Fail/Not Executed)	Pass
---------------	--------------------	-------------	--------------	------------------------------------	------

S #	Prerequisites:
1	Access to a Browser
2	
3	
4	

S #	Test Data
1	Username = test
2	Email = test@gmail.com
3	Password = test
4	First Name = John
5	Last Name = Doe
6	Role = Data Owner

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to https://dashboard-eu-onenet.euprojects.net/	Site should open	As Expected	Pass
2	Enter User Details	Credentials can be entered	As Expected	Pass
3	Click Sign up	User is logged in	As Expected	Pass
4				

8.3 Frontend Testing

The frontend testing in OneNet regards the Connector: User Interface component, which we also refer as OneNet dashboard in this document. Testing a front-end application brings some challenges that the “classic” tests do not have: you need to orchestrate a real browser. Browsers are heavy applications by definition, and you need to launch them, manage them through a specific library, leverage some APIs to automate the same kind of interactions the user would do, and then check if the state of front-end application (essentially what it shows) is the same as you expect it to be. Everything is asynchronous: the user-simulated interactions are asynchronous, the UI reacts asynchronously, the browser reacts asynchronously, the tool you use to orchestrate and communicate with the browser is asynchronous.

To automate and test a front-end application you normally need two different tools:

- A test runner: the one that takes care of executing the test itself
- A browser automation tool: something that exposes some APIs to interact with a launched-on purpose browser

Cypress is a modern all-in-one end-to-end testing framework and is the chosen tool for testing OneNet Dashboard. Cypress is executed in the same run loop as our application. Behind Cypress is a Node.js server process. Cypress and the Node.js process constantly communicate, synchronize, and perform tasks on behalf of each other. The installation is very simple as it is another npm package for the OneNet Dashboard Angular frontend.

More specifically, each dashboard testcase will try to cover following steps:

- 1) Visit a web page.
- 2) Query for an element.
- 3) Interact with that element.
- 4) Assert about the content on the page.

8.4 Backend Testing

In the context of backend testing, we will perform API testing to the following OneNet components:

- Connector: REST APIs (external Platform Integration)
- Connector: Context Broker
- Connector: Configuration
- Middleware: Data access policies
- Middleware: Identity Management
- Middleware: Semantic Annotation
- Middleware: Data quality
- Middleware: Logging System
- Cyber-security: Network traffic monitoring tools

- Orchestration Workbench: Service Catalogue
- Orchestration Workbench: Performance Evaluation
- Monitoring & Analytics: Administration
- Monitoring & Analytics: Analytics

Let's analyse the example of Middleware: Identity Management component. This component has a relevant requirement "Register or modify account of OneNet Participant". From D5.3 [3], table 6: cross-platform services for category 01, Authentication & Authorization, we can easily identify the need of 2 API tests:

Method	Description
Post	Data owners and delegated parties can authenticate themselves using the service of an Authentication Service Provider, for example a Customer Portal or a Local Market Platform; for new parties, an account is created
Get	Request verification of identity or validation of account (of data owner or delegated party such as FSP) through and return result of verification/ validation (Read).

These 2 testcases represent the so called sunny-day scenarios where a new account is created successfully, and the user is authenticated towards the OneNet platform. A number of provocative testcases can also be examined here like:

- User already has an active account so he cannot provide the same credentials twice to have two separate accounts.
- User's password does not follow recommended structure or number of minimum characters, so it is not accepted in the account creation process.
- User tries to login giving wrong username/password.

8.5 Integration (Frontend-Backend) Testing

In the final phase, integration testing will be performed, using Cypress framework. Because of the fact that this kind of testing covers all the three layers of the web-application, the full flow for the example of the creation of a new OneNet account is described below:

1. Participant visits the site that hosts the OneNet dashboard.
2. Participant clicks on the "Sign up" button.
3. Participant fills in the details needed in the registration form.
4. Participant clicks on "Submit" when the form is complete.
5. A check is performed in the database that acts as user administration database if the user already exists.
6. If the user already exists in the database, a rejection request is received by the participant in the form of an informative message "User already exists".

7. If the user is indeed new, a new entry is created in the user administration database with a unique `user_id`.

A test template similar to the one used in section 9.2 can also be used here so as to describe each test in an accurate and reproducible way. As we proceed on the next phases of the OneNet implementation, a full list of testcases is expected to be included.

9 Deploying the OneNet Connector

9.1 H/W and S/W Prerequisites

The hardware and operating system prerequisites are:

- A 2-core processor
- 4GB RAM Memory
- 50GB of disk space or more

The software prerequisites include:

- Centos 7 or Windows Server Operative System (OS); Docker and
- Docker-compose;

OneNet software and its components will be delivered utilizing the Docker containers functionalities. Firstly, the Docker platform has to be downloaded and installed accordingly to the OS of the server to host the deployment.

9.2 Deployment

The deployment of the OneNet Connector is possible through the software and the instructions that are available under

<https://github.com/european-dynamics-rnd/OneNet/tree/master>

Instructions for Deployment are available under

<https://github.com/european-dynamics-rnd/OneNet/blob/master/README.md>

As the development of the OneNet Connector is an ongoing process, deployment instructions might be updated. The above link will always lead to the last version, indicating possible alterations/additions in comparison with previous versions.

The connector's S/W is being deployed through containerisation in a Docker environment. It can be deployed either under a Windows or a Linux server.

Thus, the basic steps towards deployment (as they are in detail documented in the deployment guide) are:

- Install the Docker environment on a Windows or Linux Server
- OneNet Containers installation on Docker
- Central User Interface registration/configuration
- Network configuration

9.3 GUI Deployment

The OneNet connector's GUI is already deployed centrally for the 1st release to allow seamless corrections and adaptations towards a finalised version. Thus, users will not need to re-deploy for every change which will occur.

For the next releases part of the development effort will also be focused in having the connector's GUI to be deployed locally as an inextricable part of the OneNet connector

10 References

- [1] D5.1 – OneNet concept and requirements, Sep. 2021 ([link](#))
- [2] D5.2 – OneNet Reference Architecture, Sep 2021 (confidential)
- [3] D5.3 - Data and Platform Assets Functional Specs and Data Quality Compliance, Nov. 2021 (confidential)
- [4] D5.4 - AI, Big Data, IoT Enablers and FIWARE compliant interoperable interfaces for grid services, Jan 2022 (confidential)
- [5] D5.5 - Report on Technical specifications for data models/platform agnostic middleware, Jan. 2022 (confidential)
- [6] D5.6 - Report on Extended Data, Platform and Service Interoperability, Mar. 2022 (confidential)
- [7] [D5.8 Report on Cybersecurity, privacy and other business regulatory requirements](#) ([link](#))
- [8] D6.7 - Compliance to Reference Architecture Management Tools, Dec. 2021 ([link](#))
- [9] Horizon 2020 - Work Programme 2018-2020 Secure, clean and efficient energy ([link](#))
- [10] Spring Boot ([link](#))
- [11] MongoDB ([link](#))
- [12] IDS based TRUE (TRUsted Engineering) Connector ([link](#))
- [13] FIWARE Orion Context Broker-LD ([link](#))
- [14] <https://internationaldataspaces.org/wp-content/uploads/IDS-Reference-Architecture-Model-3.0-2019.pdf>

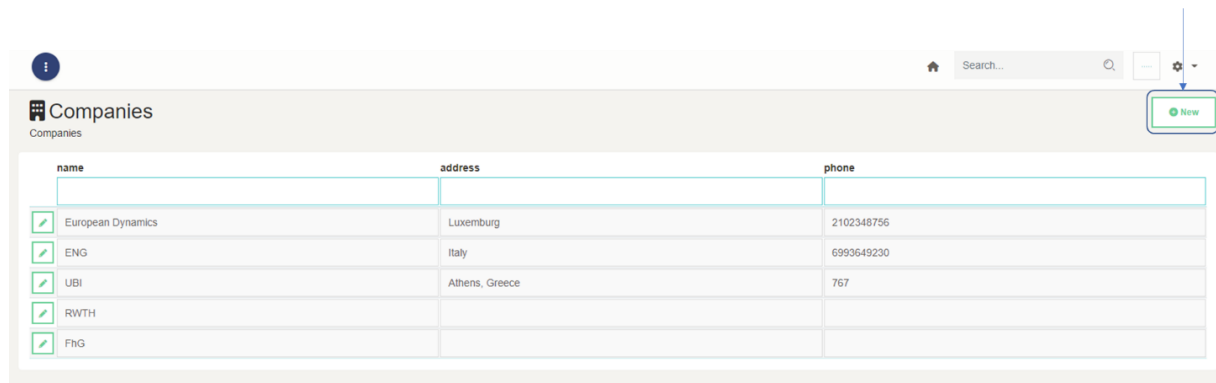
11 ANNEX – Graphical User Interface (GUI) Manual - v1.0

1 General overview

The design and development of the GUI interface adopted the approach to streamline the participation of OneNet users into the OneNet data ecosystem, supporting and most importantly promoting the seamless and secure data exchange amongst them in a decentralized manner. The sole requirement is the deployment of a OneNet connector at the premises of the new entrant. As the OneNet connector bases its implementation on Fiware True Connector (FTC) which enables trusted data exchange in order to be active part of an IDS principles. The proposed GUI essentially decouples the need for users understanding of the FTC's/ IDS processes and the corresponding information models.

2 System administrator notes

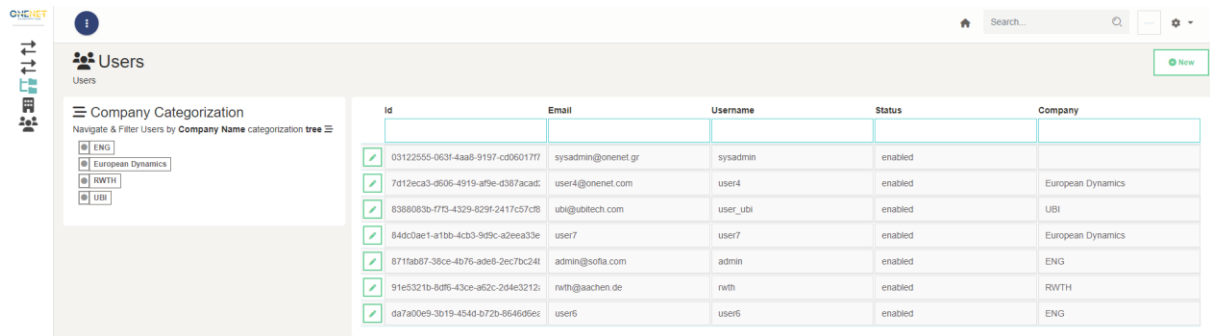
The system administrator is essentially responsible for the administering the operation of the OneNet central Middleware. For the sake of the first release, the creation of new OneNet users shall take place solely through the system administrator. A system administrator might create new user profiles which always have to be assigned with an affiliated Company. Screen 1, illustrates the respective dashboard in which the system administrator might add or edit companies in the ecosystem.



	name	address	phone
<input checked="" type="checkbox"/>	European Dynamics	Luxemburg	2102348756
<input checked="" type="checkbox"/>	ENG	Italy	6993649230
<input checked="" type="checkbox"/>	UBI	Athens, Greece	767
<input checked="" type="checkbox"/>	RWTH		
<input checked="" type="checkbox"/>	FhG		

Screen 1: Companies registry.

The corresponding dashboard for user management (configuration of existing and creation of new users) is presented in Screen 2. Multiple users might be assigned with a company.

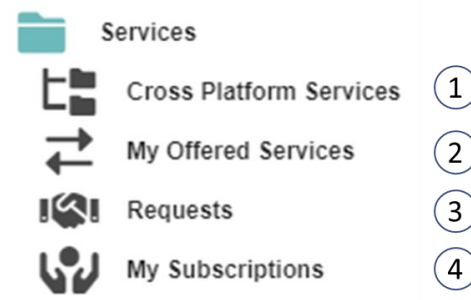


id	Email	Username	Status	Company
03122555-063f-4aa8-9197-cd060177f	sysadmin@onenet.gr	sysadmin	enabled	
7d12eca3-d606-4919-af9e-d387acac	user4@onenet.com	user4	enabled	European Dynamics
8388083b-f7f3-4329-829f-2417c57c8	ubi@ubitech.com	user_ubi	enabled	UBI
84dc0ae1-a1bb-4cb3-9d9c-a2ee833e	user7	user7	enabled	European Dynamics
871fab67-38ce-4b76-ade8-2ec7bc24f	admin@sofia.com	admin	enabled	ENG
91e5321b-8df6-43ce-af2c-2d4e3212f	rwth@saachen.de	rwth	enabled	RWTH
da7a00e5-3b19-454d-b72b-864606e	user6	user6	enabled	ENG

Screen 2: User Management.

The system administrator might view from Services tab (Screen 3):

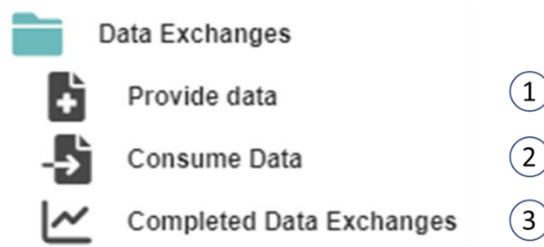
1. any information related cross-platform services and update this information. Its system's administrator sole responsibility to add or update a new cross-platform service. For instance, a system administrator might be wish to update the semantic definition of a cross-platform service,
2. all offered services from all users,
3. all requests sent to services providers to accept/reject a service subscription,
4. any service subscriptions to any cross-platform service.



Screen 3: Services dashboard and sub-menus

The system administrator has also the capability to view meta-data information about OneNet data exchanges (Screen 4):

1. any data provisions from OneNet data providers,
2. any **meta-data** information for data available for consumption,
3. a dashboard with a list of completed data exchanges.



Screen 4: Data exchanges dashboard and sub-menus.

3 OneNet users

Prior to the description of the OneNet Connector's GUI, it is vital to highlight the key concept/work flow it is designed to operate in order provide seamless and secure data exchange among platforms. As illustrated in Fig. 1, there are certain preconditions for new OneNet entrants, to provide or consume data:

- **Data provision:** OneNet user that wishes to act as a data provider, needs primarily, to register (i.e., in the OneNet Middleware) what type of services will be offered. This is performed in centralized manner, in order to inform all the potentially interested parties about this new service offering. The service offering is assigned with cross-platform service type, as defined in OneNet, so that there is common understanding (description, syntactics/semantics). Once this new service offering is registered, then a service provider can make available data items in the OneNet ecosystem. In fact, only meta-data information will be made available (for this new data item) to all OneNet users that have an active subscription for this service.
- **Data consumption:** OneNet user that wishes to consume data, needs firstly, to express the interest for a specific cross-platform services, by making a service subscription, accordingly. Once the subscription is market accepted from the service provider, then the data consumer will be able to get updates (i.e., given proper meta-data descriptions) on new data item assigned with this service. For new data item, the data consumer can make a GET request to the provider by clicking the respective button as it is explained in the forthcoming sections.

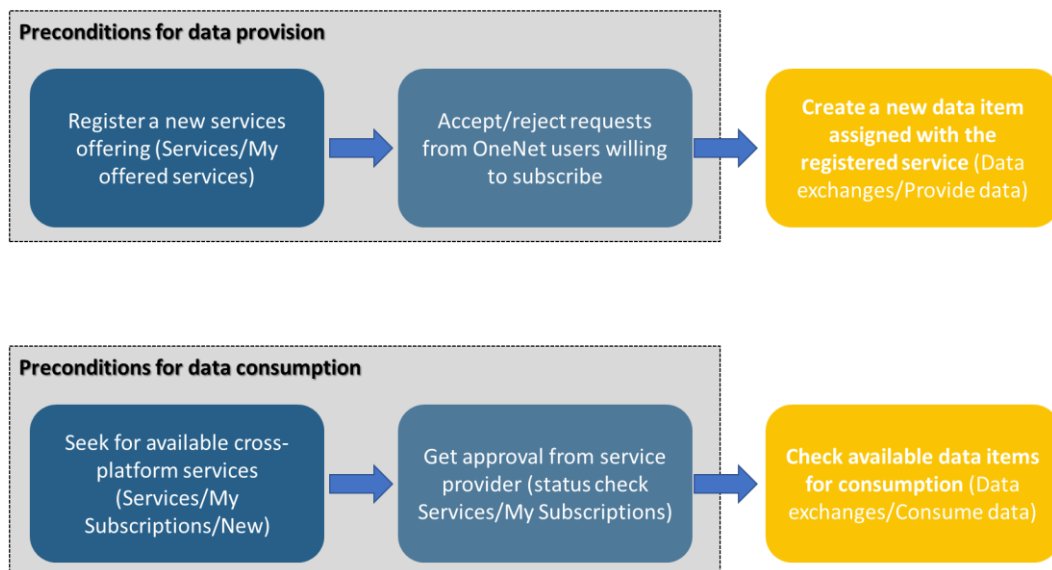
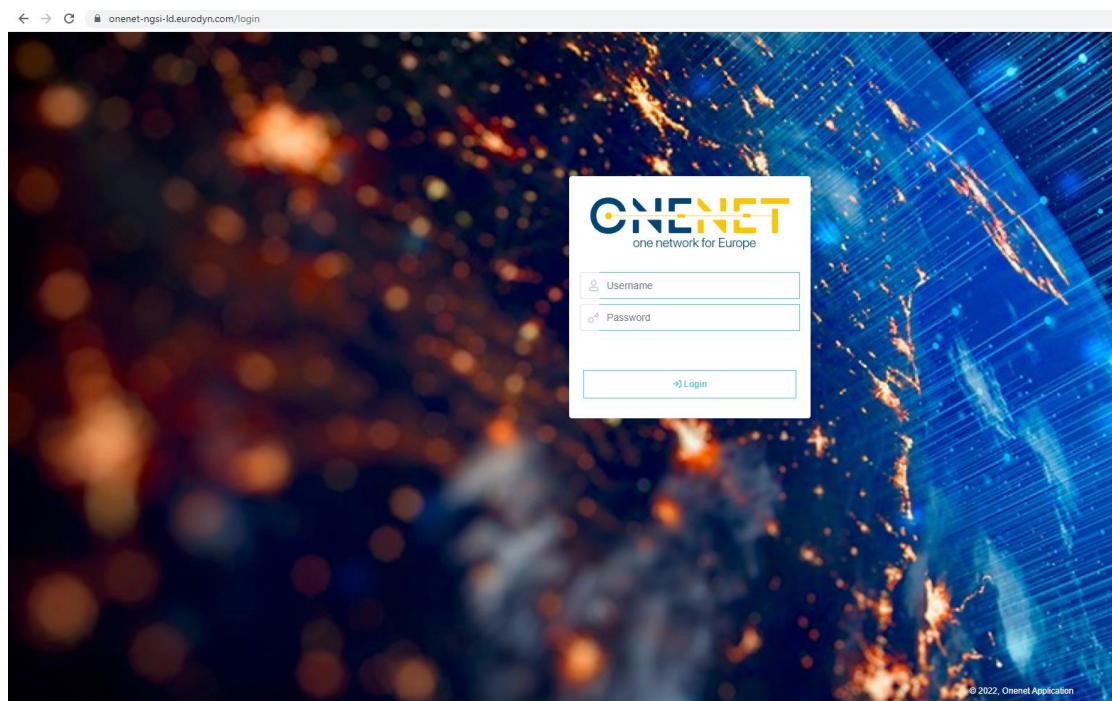


Fig. 1: Key concept for exchanging data through OneNet Connector.

3.1 Accessing OneNet GUI

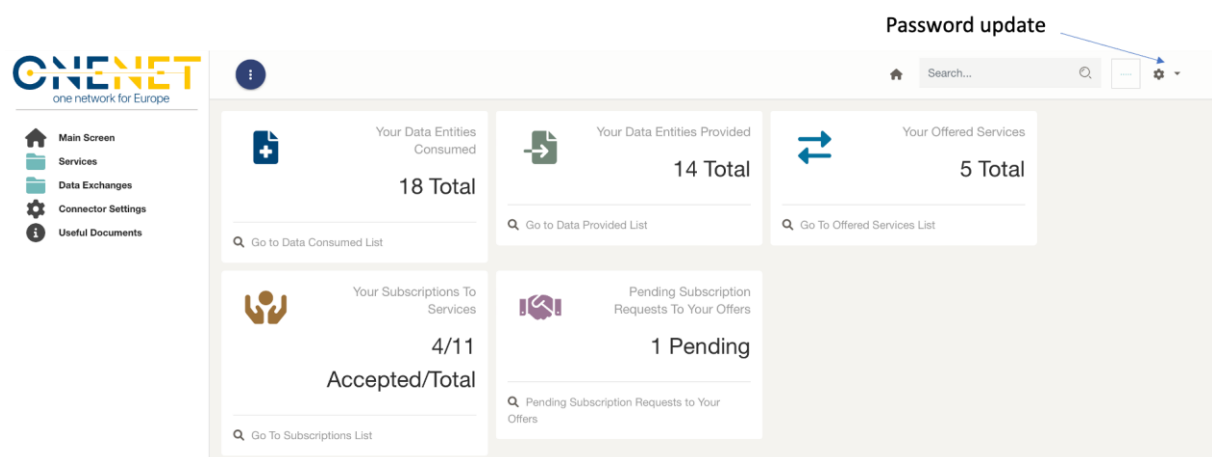
Regarding the first release, for companies/users that are willing to deploy the OneNet connector proper user profiles will be created and shared with them, which will be used to get access to the GUI of OneNet (<https://onenet-ngsi-ld.eurodyn.com/login>). The Log-in screen appears in Screen 5.



Screen 5: OneNet Connector's GUI log-in screen.

3.2 Main screen

The main screen that the user will be redirected to is presented in Screen 6, which essentially provides an overview of KPIs calculated at the OneNet connector's level. Such information might refer to data items consumed/provided, offered services, active subscriptions.



Screen 6: Main screen.

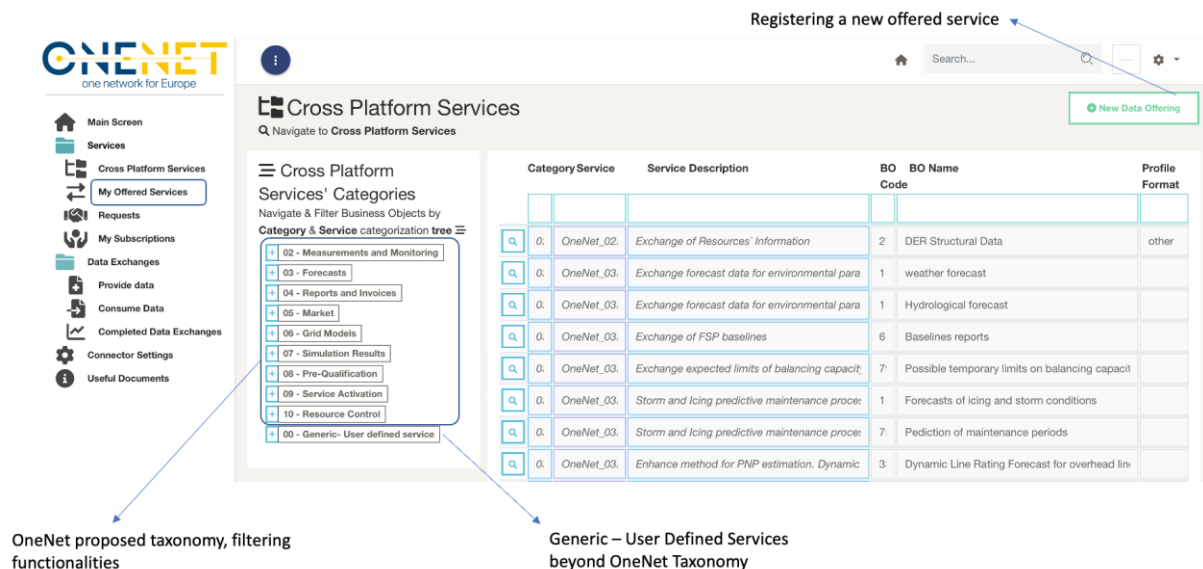
A user can update the preset password by clicking on settings as illustrated in Screen 6.

3.3 Services

In general, the services framework is designed to provide all the necessary details about the definition of OneNet cross platform services (i.e., business objects, functional description, semantic definition), the offering of a cross-platform service into the OneNet ecosystem as well as the subscription in a cross-platform service.

3.3.1 Cross-Platform Services

The cross-platform services tab is essentially a view-only environment which provides a directory on cross-platform services as they are proposed and categorized in OneNet project (see Screen 7). It should be noted that if an OneNet user cannot find a matching cross-platform, then it is indicated that the 00- Generic- Non-existing shall be used providing analytical descriptions for this cross-platform service. The latter, will ease the evolution of cross-platform services and proceed with updates on the catalogue.



Registering a new offered service

OneNet proposed taxonomy, filtering functionalities

Generic – User Defined Services beyond OneNet Taxonomy

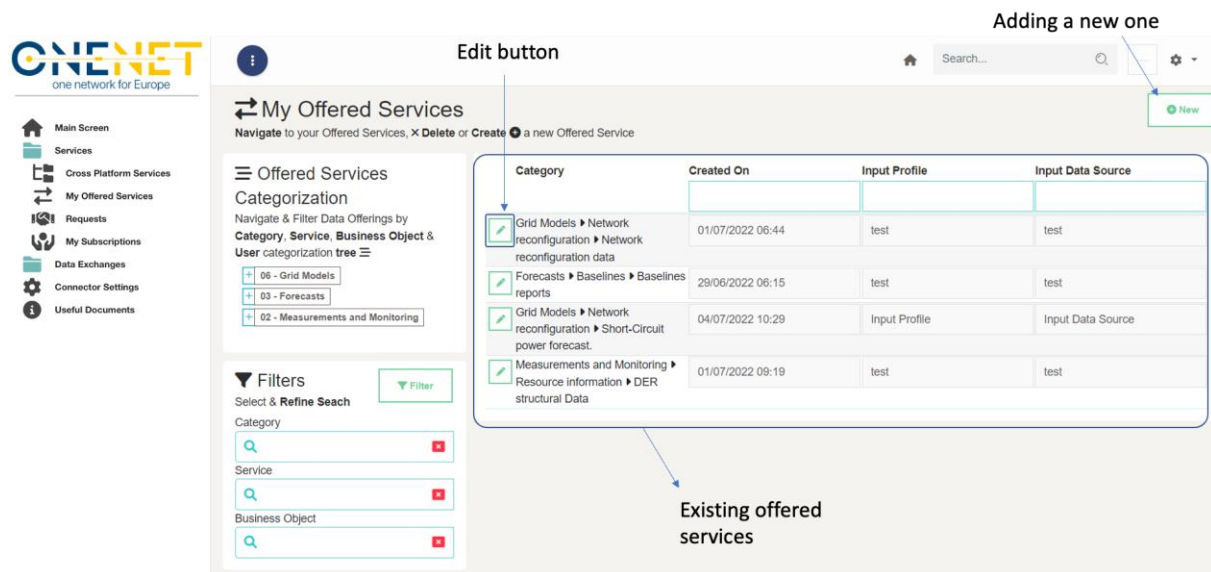
Category	Service	Service Description	BO Code	BO Name	Profile Format
0.	OneNet_02.	Exchange of Resources' Information	2	DER Structural Data	other
0.	OneNet_03.	Exchange forecast data for environmental para	1	weather forecast	
0.	OneNet_03.	Exchange forecast data for environmental para	1	Hydrological forecast	
0.	OneNet_03.	Exchange of FSP baselines	6	Baselines reports	
0.	OneNet_03.	Exchange expected limits of balancing capacity	7	Possible temporary limits on balancing capacity	
0.	OneNet_03.	Storm and Icing predictive maintenance proce:	1	Forecasts of icing and storm conditions	
0.	OneNet_03.	Storm and Icing predictive maintenance proce:	7	Pediction of maintenance periods	
0.	OneNet_03.	Enhance method for PNP estimation, Dynamic	3	Dynamic Line Rating Forecast for overhead line	

Screen 7: Cross-platform services tab.

A user by clicking into the “New Data Offering” button, moves into the next tab of My Offered Services (see next section).

3.3.1.1 My offered Services

This tab is when a user wishes to provide/deliver a new cross-platform service. By clicking this tab, see Screen 8, the user can preview existing offered services by the same user, to which one sort based on the filters. The user has the option to edit an existing service and proceed with updates on the offering, by clicking on the edit Button. For a new offering, the user needs to click New.



Adding a new one

Edit button

Search...

New

My Offered Services

Navigate to your Offered Services, X Delete or Create a new Offered Service

Offered Services

Categorization

Navigate & Filter Data Offerings by Category, Service, Business Object & User categorization tree

06 - Grid Models

03 - Forecasts

02 - Measurements and Monitoring

Filters

Select & Refine Search

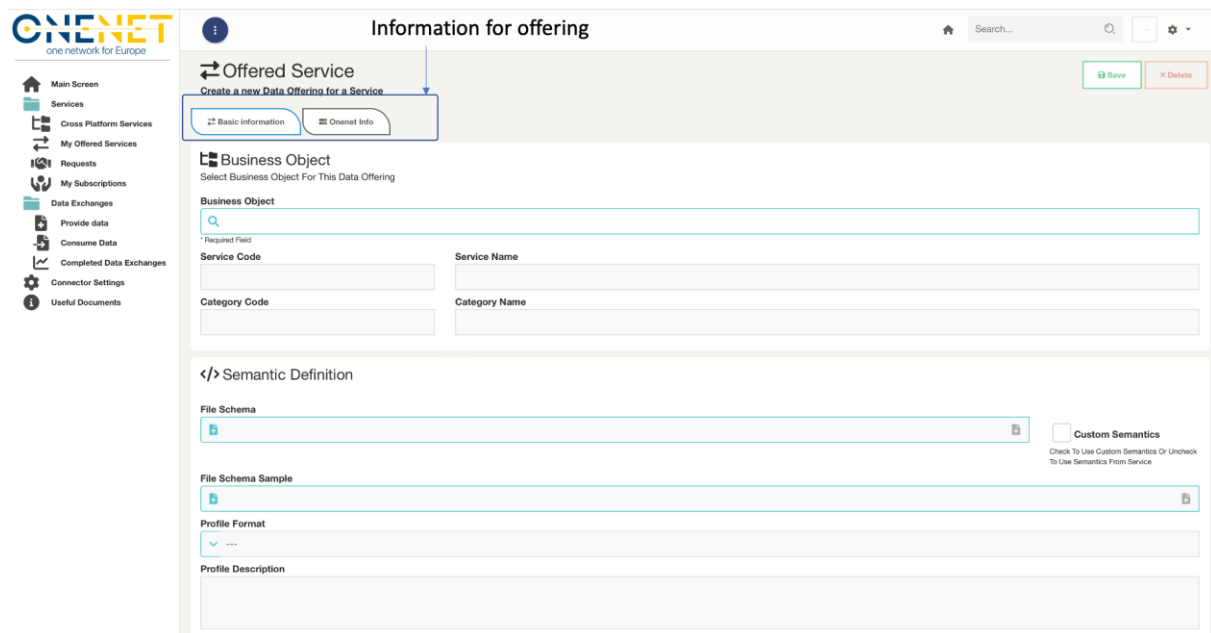
Filter

Category	Created On	Input Profile	Input Data Source
Grid Models ▶ Network reconfiguration ▶ Network reconfiguration data	01/07/2022 06:44	test	test
Forecasts ▶ Baselines ▶ Baselines reports	29/06/2022 06:15	test	test
Grid Models ▶ Network reconfiguration ▶ Short-Circuit power forecast	04/07/2022 10:29	Input Profile	Input Data Source
Measurements and Monitoring ▶ Resource information ▶ DER structural Data	01/07/2022 09:19	test	test

Existing offered services

Screen 8: My offered services tab.

By clicking New, the user will view a pop window as in Screen 9. The user has to assign this new service with a business object that is in turn under a cross-platform service. By clicking in the Business Object the list that is presented in the Cross-platform Services tab will again appear. The user has also to claim whether service delivery will occur using OneNet harmonized data profiles, else to define the data profile of information to be shared. In the OneNet info there is information regarding the service provider timestamp, the assigned by the system uuid as well the information of the service provider as in Screen 10. **Error! Reference source not found..** The semantic definition area is the one where the service provider states details on how the data to entities (assigned with this service) will be formatted. For this reason, once a user selects a specific cross-platform service then the harmonized -if any- will be loaded automatically. In case the services provider wishes to set custom profile, then, needs to check the box "Custom Semantics". Nonetheless, due to the fact that T5.6 is currently evolving on the harmonized semantic definitions, those fields are essentially locked.



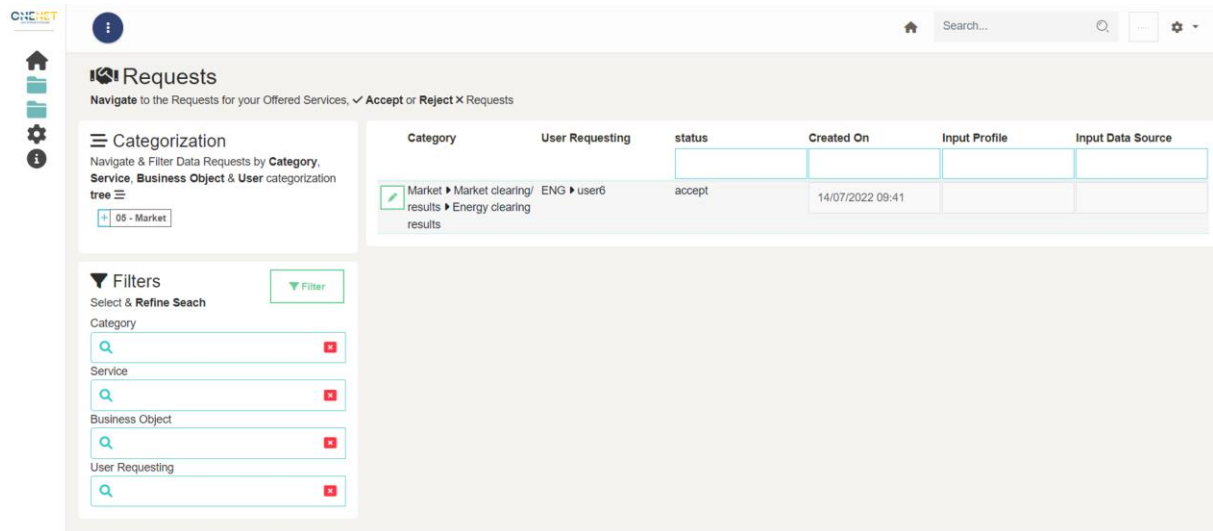
Screen 9: Adding a new cross-platform service offering.



Screen 10: OneNet Info on my offered services.

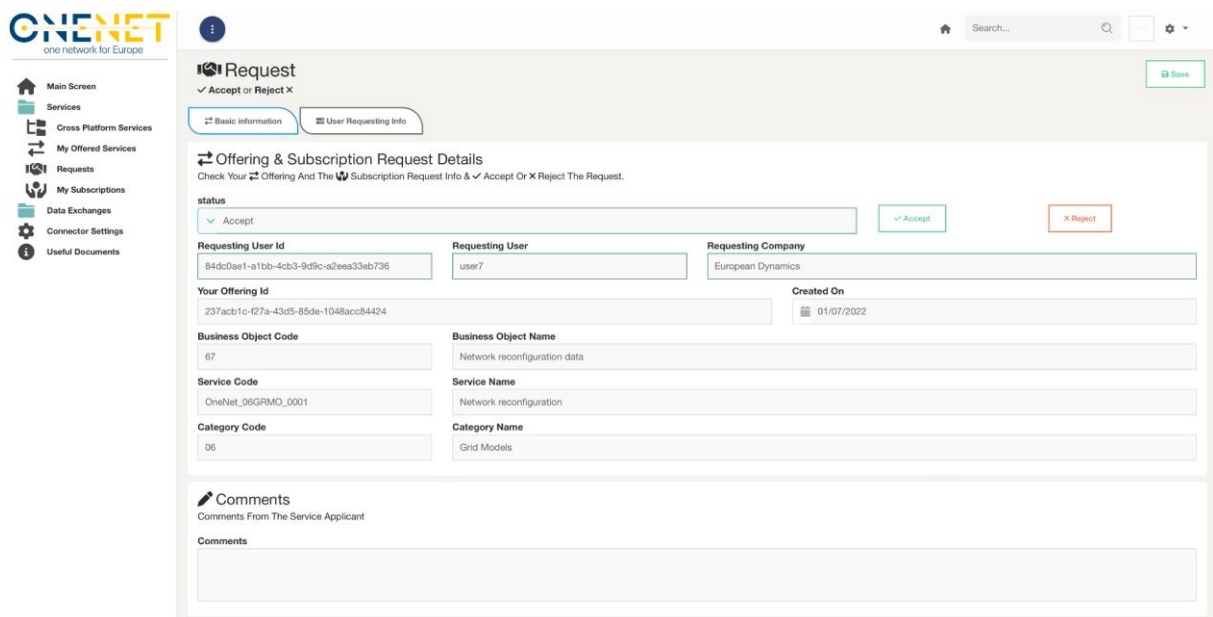
3.3.2 Requests

This tab provides a listed summary of incoming requests from other OneNet users for the Offered services, as in Screen 11. The service can view the list of pending or already responded requests. The requests can be addressed by clicking the Edit button. The window that appears accordingly is on Screen 12, which, essentially, provides information about requesting user and company. From this tab, a service can even change the status of a service from “Accept” to “Reject.” Note that, once a request is accepted or rejected then it cannot change again; this sort of limitation is set for the first release.



Category	User Requesting	status	Created On	Input Profile	Input Data Source
Market ▶ Market clearing/ results ▶ Energy clearing results	ENG ▶ user6	accept	14/07/2022 09:41		

Screen 11: Requests, incoming request for offered services.



Request
✓ Accept or Reject X

Basic information | User Requesting Info

Offering & Subscription Request Details
Check Your Offering And The Subscription Request Info & ✓ Accept Or X Reject The Request.

status:

Requesting User Id: 84dc0a81-a1bb-4cb3-9d9c-a2ea33eb736
Requesting User: user7
Requesting Company: European Dynamics

Your Offering Id: 237acbf1c-f27a-43d5-85de-1048acc84424
Created On: 01/07/2022

Business Object Code: 67
Business Object Name: Network reconfiguration data

Service Code: OneNet_06GRMO_0001
Service Name: Network reconfiguration

Category Code: 06
Category Name: Grid Models

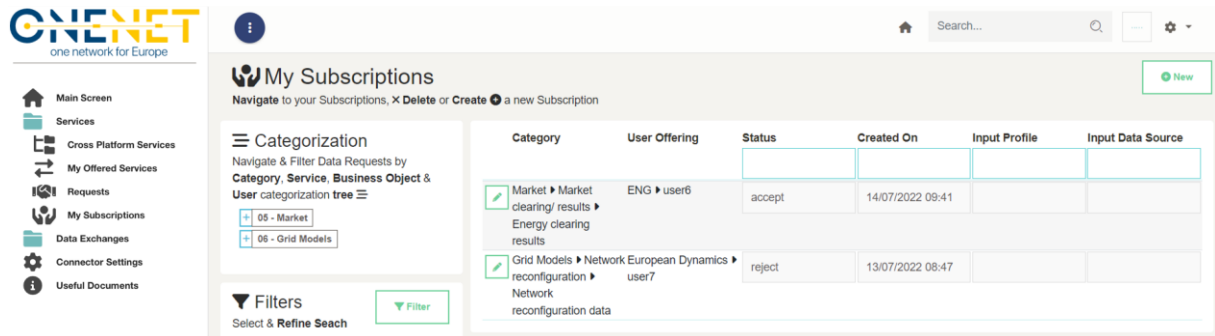
Comments
Comments From The Service Applicant

Comments

Screen 12: Responding a request on offered service.

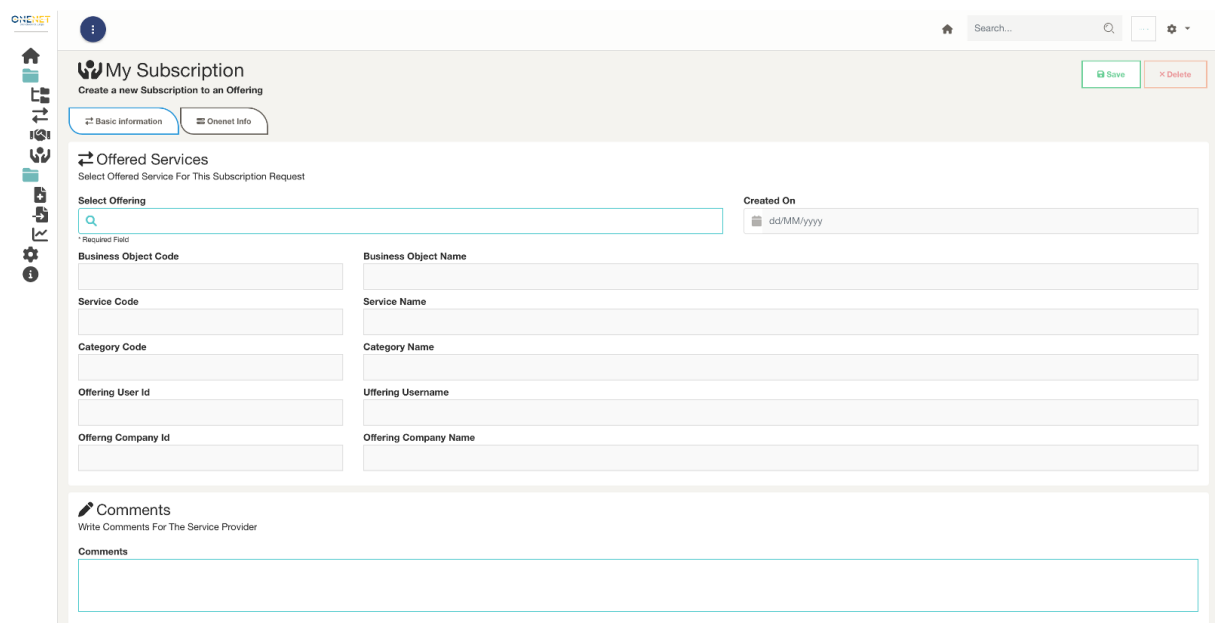
3.3.3 My subscriptions

Accordingly, the “My subscriptions” tab is for a user to make subscription to available services. The main screen on this tab is a listed summary of current subscriptions as in Screen 13.



Screen 13: My subscriptions.

A user can make a new request to service providers by clicking the “New” button, where the tab that appear is on Screen 14.



Screen 14: New subscription.

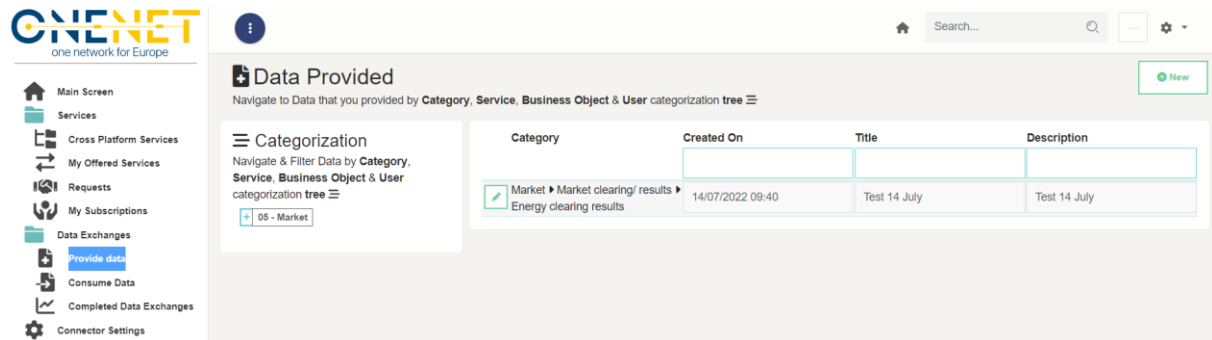
3.4 Data exchanges

The data exchanges framework is the one to perform actual data exchanges (data provision or consumption) among OneNet users serving offered services and subscriptions, accordingly. This framework is organized in three self-explanatory tab, “Provide data”, “Consume data” and “Completed Data Exchanges”.

3.4.1.1 Provide data

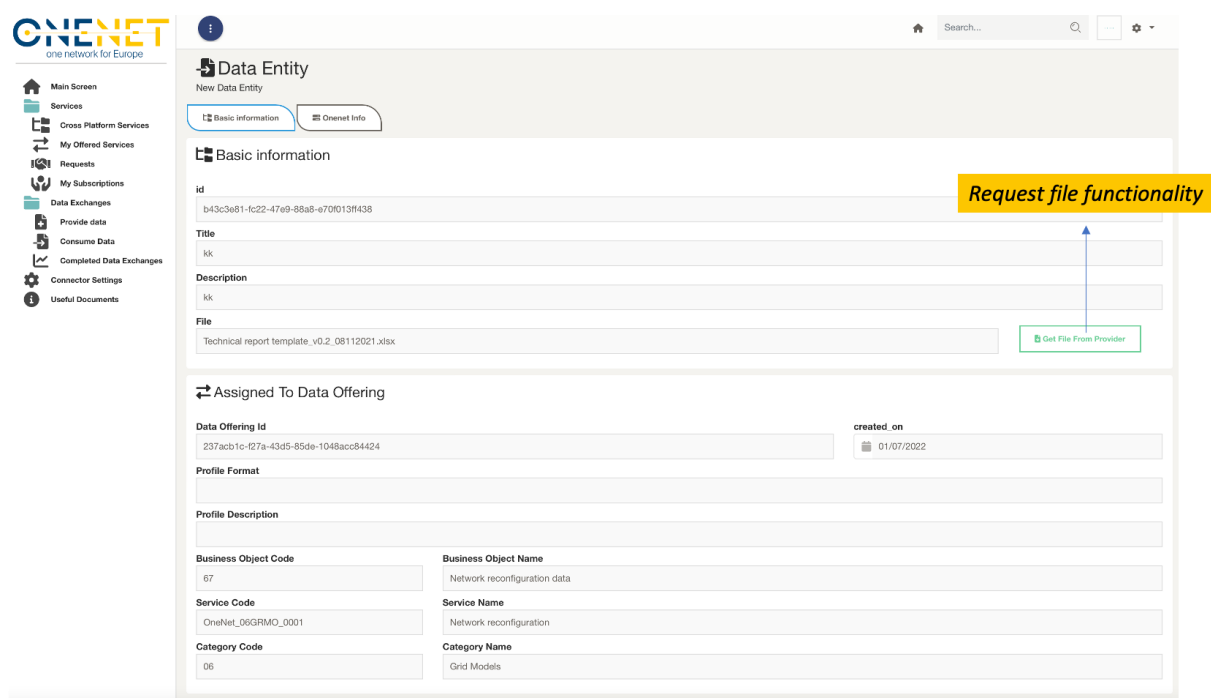
In Screen 15, it is the main Provide data tab, where it presents a list of the current user’s data item provisions. It is essential to note that for an existing data provision the file cannot be changed; fact which is prohibited for

issues related the operation of FIWARE True Connector. Therefore, the user might create a new entry for the data provision. In order to do so, the user has to click on “New”, where a window will pop up as in Screen 16.



Screen 15: Provide data tab

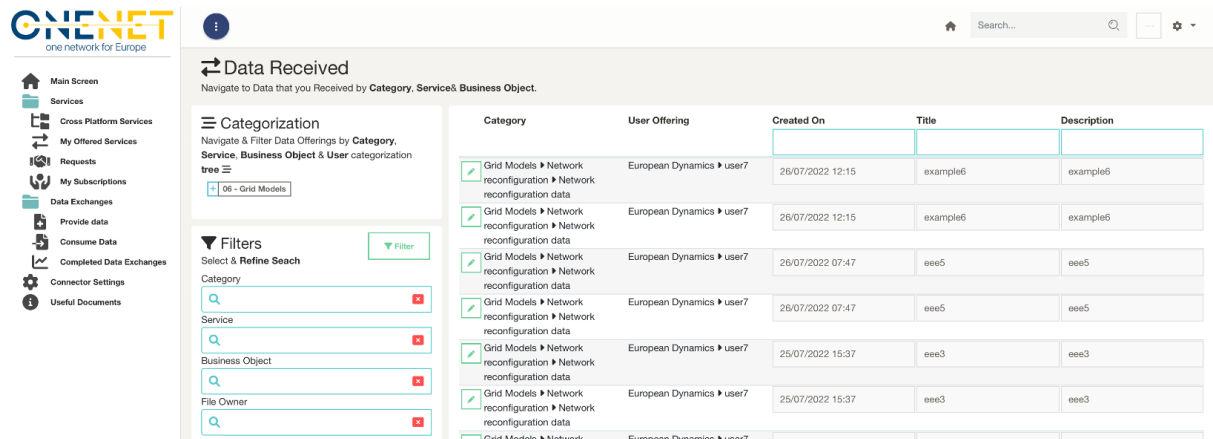
The user will have to assign the new data item with an existing offering and then upload a file through the upload file functionality. The input shall be aligned with the definition of the offered service in order the potential data consumer can utilize it.



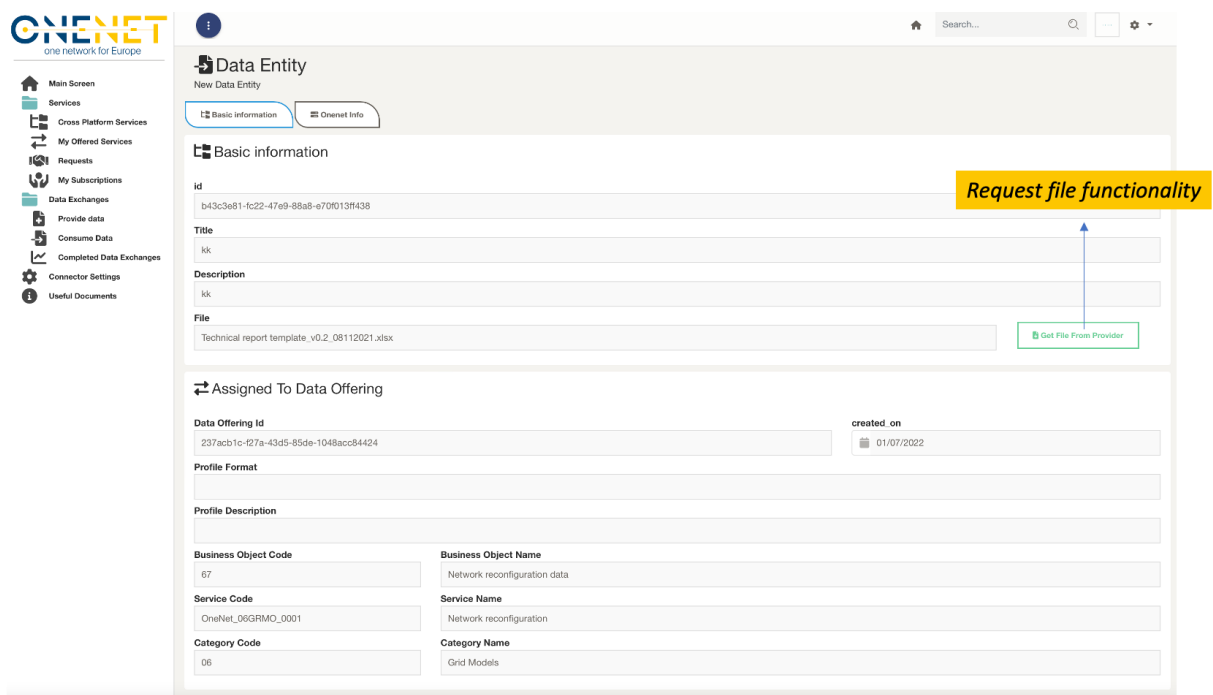
Screen 16: Adding a new data item.

3.4.2 Consume data

This tab for consuming data provides listed information for available data items (related to My subscriptions services) -see Screen 17-. To consume any of these data items, a user needs to click on the data entity and a new pop-up will appear as in Screen 18, which provides multiple meta-data information about provided data. There is this streamlined process of requesting the data by clicking a button as a download functionality, yet, the UI deals in the backend to trigger the necessary processes related to IDSA and FTCs.



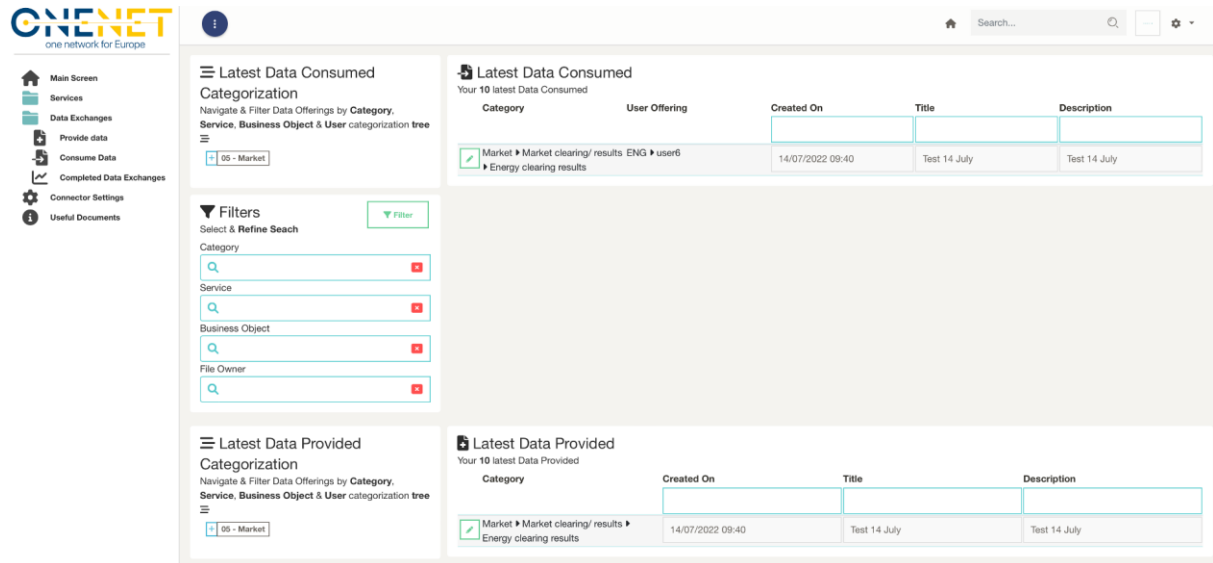
Screen 17: Consumer data tab.



Screen 18: Data entity information and get functionality.

3.4.3 Completed data exchanges

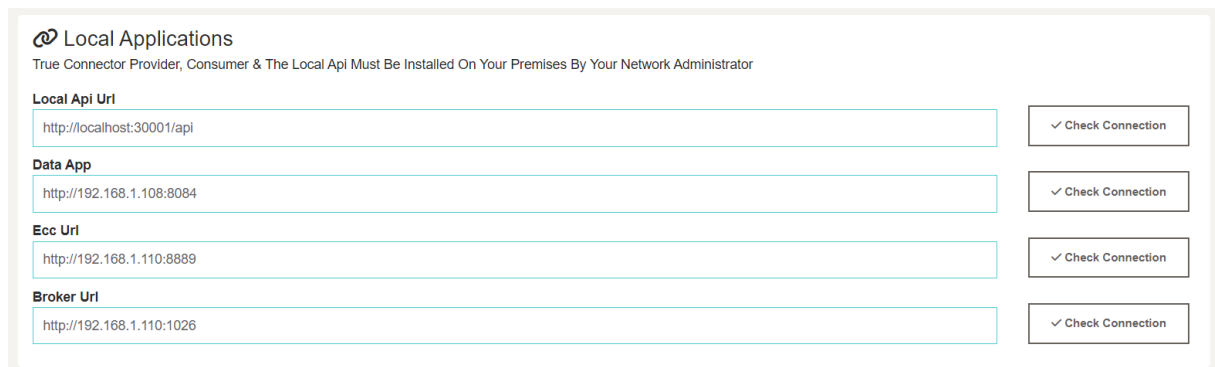
This tab provides an overview/history (see Screen 19) of completed data exchanges (data provided/consumed), where a user can check analytical information by clicking in any of those.



Screen 19: Completed data exchanges tab.

3.5 Connector Settings

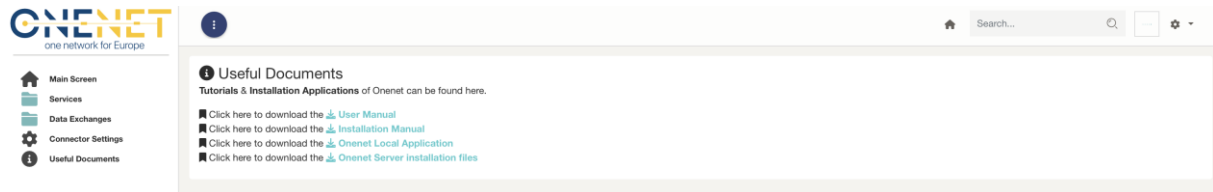
This tab is rather a configuration/setting one which provides the option to user to change the configuration files of the OneNet connector. Obviously, this can harm the connectivity and operation of the OneNet connector, therefore it is not suggested for typical (i.e., non-IT users) to test it. A user can have an overview on the actual endpoints configuration and test their connectivity (see Screen 20).



Screen 20: Connector settings.

3.6 Documents & Downloads

This tab provides useful documents (like the latest versions of the Manual and Deployment Guide) as well as for the latest S/W packages of OneNet (see Screen 21).



Screen 21: Documents & Downloads



This paper reflects only the authors' view and the Innovation and Networks Executive Agency (INEA) is not responsible for any use that may be made of the information it contains.

