FLEXERGY

Deliverable DV10 – Integration report of functional modules and lessons

Activity 5:

Software integration, functional tests, and system tests

Editor:	João Monteiro
Dissemination level: (Confidentiality)	Public
Suggested readers:	Software Engineers, System Architects, R&D Managers, Project Officer, Project Reviewers
Version:	01.00
Total number of pages:	15
Keywords:	Energy Storage, System Architecture, Energy Management,…

FLEXERGY ABSTRACT

The FLEXERGY project aims at the development of an advanced management solution, highly innovative and provided of artificial intelligence, for the management of assets of battery energy storage systems, integrated with renewable energy sources or for application within a microgrid

FLEXERGY is a project co-funded by:

Document

Name	Deliverable DV10 – Integration report of functional modules and lessons	
Document No.	AS21006298	
Revision and Date	1.0	2021-05-31
Project	FLEXERGY	
Made by	João Monteiro	
Reviewed by	Eduardo Ramalho	
Approved by	Paulo Matos	
Total Pages	15	

Language Requirements (for non-native English speakers)

In order to fully understand the content of this document, it is therefore recommended that the reader possesses a language proficiency equivalent to B1 level, according to European Language Levels

Disclosure

Revisions

Rev.	Date	Comments	Author
1.0	2021-05-31	Document release	João Monteiro

Executive Summary

This deliverable presents the report of the integration of all components of the ES Manager and further integration of software in the Demonstrator Pilot – DEMOCRAT.

The activities performed and described in this deliverable comprise:

- Functional modules integration;
- Functional tests;
- Systems tests;
- Preparation of the Demonstrator Pilot;
- Integration report.

Table of Contents

List of Figures

List of Tables

Glossary

Not applicable.

1. ES Manager

The ES Manager platform is an advanced management solution, based on artificial intelligence, for the management of power generation assets, namely renewable, engaged in a micro-grid endowed with a battery storage system. Through its intelligent management algorithms, the platform is able to optimize the performance of the micro-grid along a number of performance indicators, such as maximizing economic profit and minimizing asset degradation.

1.1 General Overview

In the context of the FLEXERGY project, the ES Manager platform was developed to use advanced artificial intelligence algorithms to optimize the performance of hybrid micro-grids integrating both renewable and conventional energy generation, battery-based energy storage, and other grid loads such as electric vehicle chargers. To this end, five main use cases were envisioned to guide the development of the system.

Use Case 1: Technical and economic optimization of a hybrid park supported by a battery energy storage system

Integrating renewable energy source in the electric grid is the main reason for the inclusion of energy storage systems, particularly battery based, since this type of technological solution is seen as a complement to integration of renewable energy sources of unreliable and intermittent characteristics, such as wind and photovoltaic generation. This use case aims to demonstrate the possibility of greater integration of renewable sources when these are coupled to a Battery Energy Storage System (BESS); in other words, envisioning a hybrid park where renewable energy generation and energy storage systems have the same point of common coupling (PCC).

Use Case 2: Battery Energy Storage Systems as support for electric vehicle charging station integration

The vast majority of electric vehicle (EV) charging stations are located in a private or semi-private environment, and as such there is a necessity to optimize and manage the EV charging process. Incorrect management of a VE charging station in tandem with widespread usage of this technology may considerably increase the peak of demand for electric power, as well as high voltage shifts, decreased energy quality and distribution losses.

Use Case 3: Holistic optimization of a micro-grid integrating BESS

The potential added value of a BESS is made obvious in the context of micro-grids with multiple DERs, including various renewable energy sources. Operational flexibility is fundamental to maximization of reliability, efficiency and resiliency that reflect a low environmental impact and increased benefits for the final consumer. BESS integration achieves such flexibility by buffering the unreliable production profiles of renewable energy sources, allowing for reduced usage of conventional energy generation methods and, consequently, their environmental impact. However, coordination between these different energy sources in real time must be assured by a high-availability, fast response management that can more easily be achieved with predictive methods based on asset generation forecasts.

Use Case 4: Maximization of net result of the storage system across provision of multiple services

To surpass entry barriers typical of BESS – such as high investment cost and short lifespan -, thorough exploration of its capability to support multiple services is necessary. Its multifunctional nature allows access to multiple sources of revenue, which gives it a high potential to overcome initial investment costs.

Economic opportunities include but are not limited to: frequency regulation; maximization of profits/minimization of costs; maximization of auto-consumption; balancing services; capacity management; and reactive energy support.

Use Case 5: Maximization of BESS lifetime through life cycle evaluation

As previously stated, short useful lifetime is one of the BESS' barriers to entry and as such of its widespread use. For example, while a typical BESS may reach its end-of-life status at the end of 10 years, a transformer may be adequate for up to 40 years with a lower maintenance cost. As such, the goal of this use case is the development of a model allowing evaluation of the BESS lifecycle, based on historical use data, to estimate the remaining useful lifetime, which itself may be used in future operations. This evaluation is made across all functions detailed in previous use cases.

1.2 Main software requirements

The FLEXERGY project aims at developing an advanced and highly innovative management solution, enabled by artificial intelligence, suitable for managing battery-based energy storage assets, integrated with renewable energy sources or meant to be applied in microgrids. This platform will be developed according to the intrinsic features of this kind of systems, being suitable for adaptation to different scopes of integration.

The functional modules developed in previous activities have the ability to comply with the requirements of the use cases defined in the project application. Not only do these modules comply in terms of algorithm, but they also comply with the flexibility that this platform requires, that is, the modules are able to host several types of assets, several types of systems and the various possible configurations between them.

In this way, the main requirement of the base platform is to manage the various functional modules and streamline the interaction among them.

Thus, during the development phase, it was developed state machines that streamline the interaction of the various modules in a microservices perspective. The modules do not depend directly on each other but are highly dependent indirectly, i.e., for example, the dispatch optimization module needs the most recent PV forecasts and the PV forecast module needs the most recent weather forecasts. With this, this activity is extremely important because it is where adjustments are made to the state machines supporting the modules and the integration of the various modules into the base platform is validated.

2. Base platform modules integration

This section will cover the integrations of the various modules that were developed directly into the base platform by performing a series of stress and integration tests.

2.1 DataBase

The database is the core of the entire platform. The database is where all the existing information in the implemented systems is found, from configuration data of the various assets or user configuration to consumption time series or weather forecast time series.

During the development phase the performance of the database developed was guaranteed and the performance of the communication between the platform and the database was also guaranteed.

The database is considered the core of the entire platform because the functional modules do not communicate with each other but interact through the database and in this way the separation between the various modules is guaranteed. Thus, within the scope of this activity a series of tests were developed to validate the integration between the state machines supporting the functional modules and the results are presented in the following table.

Table 1 – Tests by functional module.

Functional Module	Data Writing	Data Reading
Communications Platform	☑	☑
Weather Forecast API	☑	☑
PV Forecast API	☑	☑
Load Forecast API	☑	☑
Dispatch Optimization API	☑	☑
Frontend Module	☑	☑

In these integration tests were evaluated KPIs: correct writing/reading and writing/reading speed; to validate the two types of actions in stress scenarios.

In summary, with the tests performed, the functional modules write and read correctly into the database and with a processing time low enough for the functional modules to interact in a coordinated way with each other.

2.2 Comms Platform

The communication platform contains all systems that manage communication using industrial protocols such as ModBus, IEC 61850 and IEC 60870-104. The added value prediction and optimization algorithms of the platform require both a means to obtain field data which is fast, reliable, persistent and supports high throughput; and a means to communicate the calculated optimized setpoints to the active assets of the micro-grid. The equipment that receives the setpoints is commonly the same that provides field data, and often use industrial-type protocols with simplified frames (low overhead and minimalistic payloads for faster communication and higher throughput).

As such, development of the Comms Platform must guarantee that the innate characteristics of fast, reliable and frequent communication are preserved, while also being able to operate many different clients – and consequently many different assets – with little or no parallel impact.

Preliminary testing of libraries for use with the platform revealed their suitability as single clients, as they performed with very good communication times and ranged all the basic operations. However, due to time constraints and the fact that the ModBus protocol was present in all of the field assets used in our demonstration environment, the remaining protocols (IEC 61850 and IEC 60870-104) were relegated in favor of development of the ModBus protocol.

Follow-up testing performed a high stress test with multiple clients. To dismiss suspicions that clients from the framework used a common event queue to send messages, the platform was submitted to the following test: the platform would generate 28 different clients, each with 4 orders to be executed every 200ms, for 50 cycles: Write 50 Holding Registers, Write 100 Coils, Read 50 Input Registers, and Read 100 Digital Inputs.

Testing conditions put expected conclusion time at slightly over 10 seconds, assuming the system did use separate queues for every client, as a) the platform was in the same machine as the test servers, which reduces communication and transmission time, but b) the platform was running in a virtual machine with lesser-than-average available CPU power, which increases execution time in general. Real execution time for all clients variated between 11.3 and 12.5 seconds, with an average time of 11.6 seconds, confirming that each client had in fact a dedicated queue. Further technical tests were not performed due to improper testing conditions.

3. APIs integration

This section describes the integration tests of the various functional modules that were developed through APIs. During the development phase, the communications between the APIs and the base platform were already guaranteed through a REST interface. During the database integration tests the database reads and writes have also been guaranteed.

Thus, what remains in the integration phase of the various APIs are the adjustments in the state machines between the various APIs. In this way, we started the coordination of the various APIs by the APIs in which their supplied data is closest to the data to be sent to the field. In this way we have the following order:

1. Dispatch Optimization API
2. PV Generation Forecast API and Load Forecast API
3. Weather Forecast API

Since ES Manager is prepared to support various time intervals, it was necessary to develop a module that manages the timings of requests to various APIs. This module has the ability to understand when to make the various requests through automatic learning. Depending on the number and type of assets present in the system and depending on the type of active functionality, the module recreates several extreme scenarios in order to ensure the best results, both in terms of speed, security and performance.

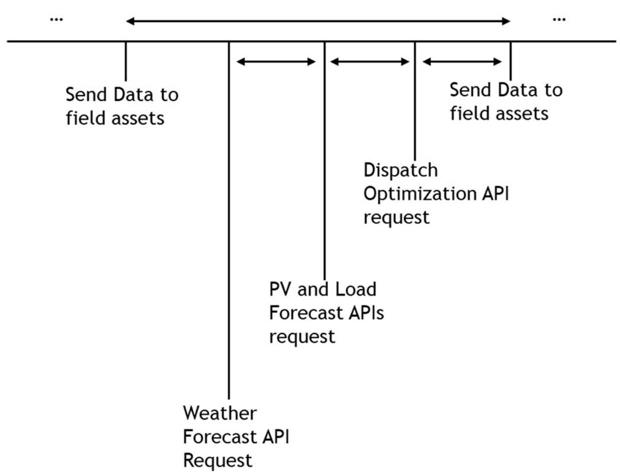

Figure 1 - APIs requests timeline.

The development of the coordination module satisfied the development team as the point of coordination between the various APIs was a critical point in the project. It was classified as a critical point because the APIs need the latest data and because it was always preferred to disaggregate the various modules in order to increase the reliability and security of the system, but also the continuous development of the platform.

Thus, with the various systems that ES Manager can accommodate, the development of this module allows that it is not necessary to adjust the various state machines from system to system.

With the degree of coordination achieved, it can be said that the various modules are integrated into the base platform and ready to be tested in the field.

4. Demonstrator Pilot - DEMOCRAT

Based on the defined test plan and, in particular, based on the use cases to be tested and validated, the work was done in order to analyze and define the appropriate topology for the test platform that targets all the identified requirements, of which the diagram in Figure 2 is an example, considering the incorporation and interconnection of the different assets that will be considered for the ES Manager's global strategy.

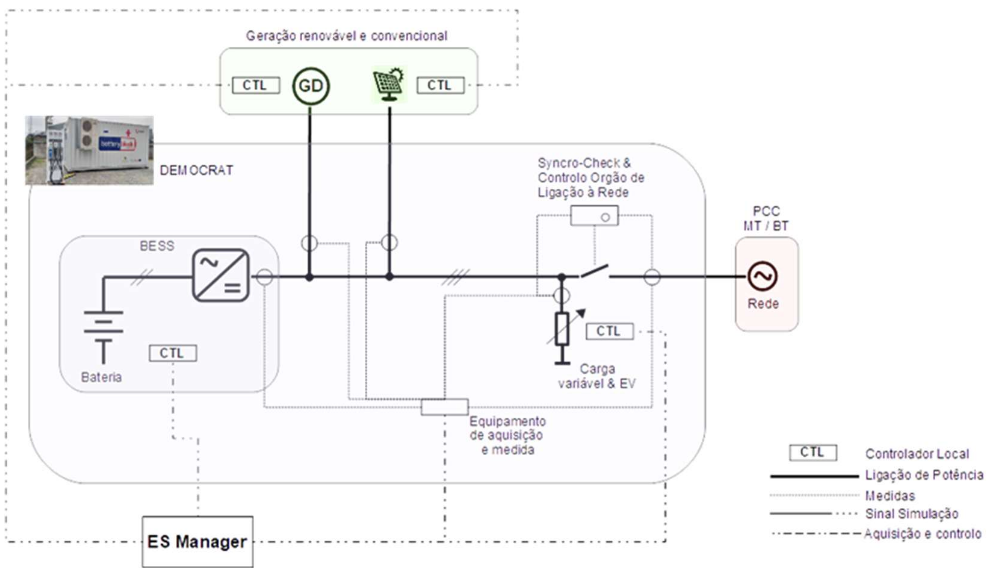

Figure 2 - Test setup – demonstration pilot.

Following the topology definition, measuring equipment, diesel generators, and other complementary equipment were purchased and rented. Next, the equipment was received and integrated according to the defined setup.

Due to the DEMOCRAT project, the integration of the various ended up not being a demanding activity due to the high flexibility and modularity that the system has, being able to have multiple configurations. Figure 3 shows the final state of the demonstration pilot.

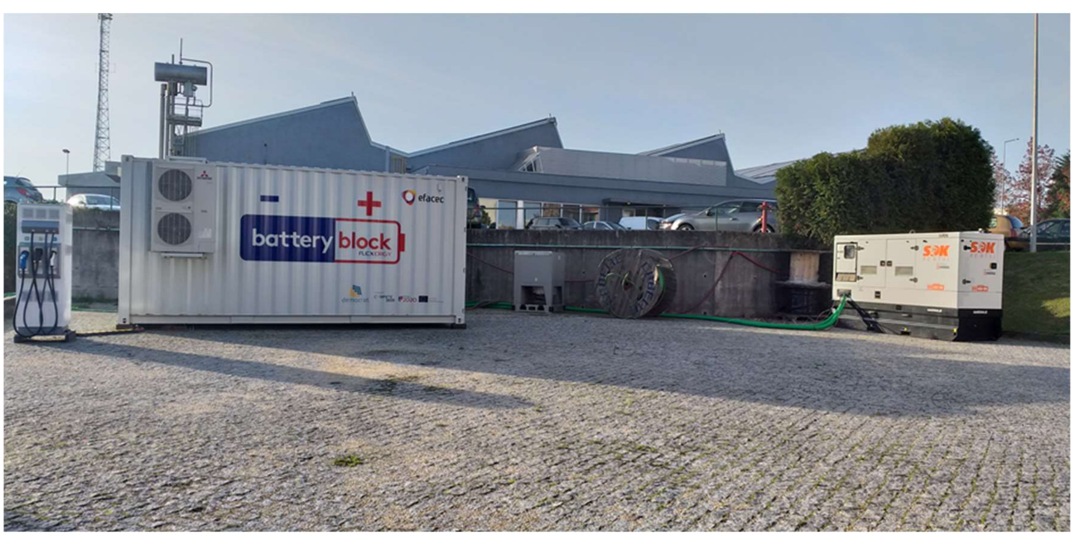

Figure 3 – DEMOCRAT demonstration pilot.

The DEMOCRAT consists of:

- BESS of 218kW / 218kWh;
- Diesel Conventional Generator of 250kVA;
- Load Bank of 250kVA;
- EV Charger of 45kVA.

Given the impossibility of having integrated a photovoltaic plant and a wind turbine at the time of reporting, the measurement data are simulated.

Thus, with this demonstration pilot it is possible to study the proposed use cases and validate the developed platform.

After the equipment was properly connected and integrated, the field controllers were prepared. The development of the field controllers' logic is intended to be able to interpret the data transmitted by the ES Manager. That is, these controllers at the level below ES Manager are intended to act as an interface between ES Manager and the assets.

For this pilot, the controllers communicate with the ES Manager and the assets via the industrial ModBus protocol. So, after the controllers were developed, several communication tests were performed. This step follows on from task 5.1. In this way, a battery of tests was performed to validate the communications for 28 clients and the following points were evaluated:

- Write 2HR, Read 50HR, Read 100DI each 200ms for 50 cycles – 11.1s;
- Write 50HR, Write 100DI, Read 50HR, Read 100DI each 200ms for 50 cycles – 11.3s.

These stress tests were given as successfully completed.

The load bank scenarios had to be simulated using typical industrial park profiles. The market data is from MIBEL, it was inserted into the database and is dated 2020.