This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 773715

Grant Agreement No.: 773715

Project acronym: RESOLVD

Project title: Renewable penetration levered by Efficient Low Voltage Distribution grids

Research and Innovation Action

Topic: LCE-01-2016-2017

Next generation innovative technologies enabling smart grids, storage and energy system integration with increasing share of renewables: distribution network

Starting date of project: 1st of October 2017

Duration: 36 months

D5.3: Report on SW installation with feedback to activities related to grid actuation, network sensing, platform integration and interoperability with legacy systems.

Organization name of lead contractor for this deliverable: ICOM	
Due date:	31/07/2020
Submission Date:	31/07/2020
Primary Authors	Viktor ALKALAIS (ICOM), Isidoros KOKOS (ICOM)
Contributors	ICOM, UdG, EYPESA
Version	Version 1.0

Dissemination Level		
PU	Public	X
CO	Confidential, only for members of the consortium (including the Commission Services)	

DISCLAIMER

This document reflects only the author's view and the Agency is not responsible for any use that may be made of the information it contains.

Deliverable reviews

Revision table for this deliverable:		
Version 0.9	Reception Date	28/07/2020
	Revision Date	30/07/2020
	Reviewers	Joaquim Melendez(UdG), Luisa Candido (EYPESA)

Contributions of partners

Description of the contribution of each partner organisation to the work presented in the deliverable.

Partner	Contribution
UdG	Participation in deployment validation. Reviewer of report.
UPC	Participation in deployment validation.
SIN	
JR	
ICOM	Participation in deployment validation. Leading author.
EYPESA	Participation in deployment validation. Reviewer of report.
CS	

Table of contents

Acronyms and abbreviations

API	Application Programming Interface
BMS	Battery Management System
CEF	Critical Event Forecaster
CGMES	Common Grid Model Exchange Specification
CIM	Common Information Model
DAP	Data Analytics Platform
DB	Database
DCU	Data Concentrator Unit
DMS	Distribution Management System
DSO	Distribution System Operator
EF	Energy Forecaster
ESB	Enterprise Service Bus
EQ	Equipment Profile
GIS	Geographic Information System of CGMES
GL	Geographical Location profile of CGMES
GM	Grid Meter (meter installed at DCU)
GOS	Grid Operation Scheduler
HTTP	Hypertext Transfer Protocol
ILEM	Intelligent Local Energy Manager
JSON	JavaScript Object Notation
LV	Low Voltage
MDC	Meter Data Collector
MDMS	Metering Data Management System
PED	Power Electronics Device
PFS	Power Flow Simulator
PMU	Phasor Measurement Unit
PQM	Power Quality Monitor
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SG	Smart Grid
SM	Smart Meter
UI	User Interface
WAMS	Wide Area Monitoring System
XML	eXtensible Markup Language

Executive summary

RESOLVD solution is composed of a complex software (SW) system that spans across multiple modules, involving data exchanges with multiple sources as well as algorithmic and data transformation functions. The objective of this report is to document the methodology followed during the software deployment and testing in realistic conditions It also indicates he results of the testing and the lessons learnt, i.e. the issues faced along with the mitigation actions identified.

The testing regimen followed during deployment covered different layers: that of integration testing, focusing on how the different components of the system are working together and interact with the external environment; end-to-end testing of the whole system, simulating real-world test conditions. The focus area of this work concerns the software components of the RESOLVD solutions, i.e. the Operation Applications, the Supervision and Analytics services, the Enterprise Service Bus (ESB) and the Data Analytics Platform (DAP) as documented in the architectural design of *D1.3 "Interoperability and Integration Analysis and Requirements"* and *D4.1 "Detailed description of the platform"*. Nevertheless, the test plan also involved interaction with other parts of the RESOLVD solution such as the Power Electronic Device (PED) and the Wide Area Monitoring System (WAMS), as well as with legacy systems of the DSO (i.e. MDMS, SCADA) and external systems (i.e. weather services).

Detailed test cases were designed to validate the proper function of the software components, detailing – apart from the test name and description - the participating modules (test items), the test area and the dependency to other test cases.

Test cases were conceptually decomposed in the following areas:

1. **Accessing field sensing and external data**: The focus of these test cases is on data retrieval/receipt from various field' sensing devices as well as from the weather services.
2. **Integrating analytic services**: Concerns that integration of the various analytics modules. These test cases have a dependency on the test cases of area 1 as well as on test cases of the same level.
3. **Dispatching grid actuation commands**: Concerns the dispatch of command to the field devices. These cases have a dependency from test cases of area 1 and area 2.
4. **Operating the system through the user interface**: Tackles the end-user viewpoint, enabling the end-to-end testing of the solution, since the user interface integrates information for the execution of the whole business workflow.

Mocked input was utilized to simulate real-world scenarios, when not feasible otherwise.

The identified issues mostly concerned data quality and availability. For these cases, mitigation actions were identified for the different stakeholders: from the perspective of the analytics provision, a more robust design was devised for tackling data issues; from the perspective of integration, a process for continuous monitoring of the health of incoming data was implemented; from the perspective of the end user and data provider, the importance of selecting the appropriate data sources for such services was highlighted.

The work documented in this report, along with the work documented in deliverable *D5.2 "Report on HW installation with feedback to activities related to grid actuation, platform integration and interoperability with legacy systems"*, aims to prepare the validation environment for transition in the piloting phase of the project.

1. Introduction

1.1. The RESOLVD project

The objective of RESOLVD project is to improve the efficiency and the hosting capacity of distribution networks, in the context of highly distributed renewable generation by introducing flexibility and control mechanisms in the low voltage grid.

An innovative advanced power electronics device, with integrated storage management capabilities provides both switching and energy balancing capacities to operate the grid optimally. Continuous power flow control between storage and the grid, and also between phases, results in a flatter and reduced demand curve at the substation level with an associated loss reduction and an improved voltage control and quality of supply.

The enhanced observability of RESOLVD, provided through cost-effective PMUs and state-of-the-art short-term forecasting algorithms that predict demand and renewable generation, permit a reduction of uncertainty in grid operation and an increased efficiency. RESOLVD proposes hardware and software technologies to improve low voltage grid monitoring with wide area monitoring capabilities and automatic fault detection and isolation.

These improved observability and monitoring technologies, combined with the capability of enhanced actuation of grid elements benefits from robust scheduling methods that support self-healing and grid reconfiguration, hence allowing efficient grid operation and a maximised renewable hosting capacity.

1.2. Scope of the reported work

RESOLVD solution is composed of a complex software system that spans across multiple modules, involving data exchanges with multiple sources as well as algorithmic and data transformation functions. The objective of this report is to document software integration, deployment and pre-pilot testing issues related to the demonstration scenario. It aims to present the methodology, criteria and lessons learnt from the field integration and testing phases.

The content of this report is of interest for technical staff (e.g. software developers/testers) providing an overview of the procedure followed for testing the deployment of the solution on the field, but also to a broader audience who seeks information for similar experiences in the Smart Grid (SG) domain.

1.3. Report structure

This section summarises the work presented in the remaining chapters of the report:

- Chapter 2 provides a brief overview of the RESOLVD solution and details the focus area of the current report with regards to deployment and testing;
- Chapter 3 presents the methodology followed for testing the deployed software;
- Chapter 4 documents the results of the deployment process and relevant experiences, focusing on the quality of data received from field devices;
- Chapter 5 presents the conclusions and next steps.

2. System overview

The following figure presents a high-level architectural description of RESOLVD solution [1], [2].

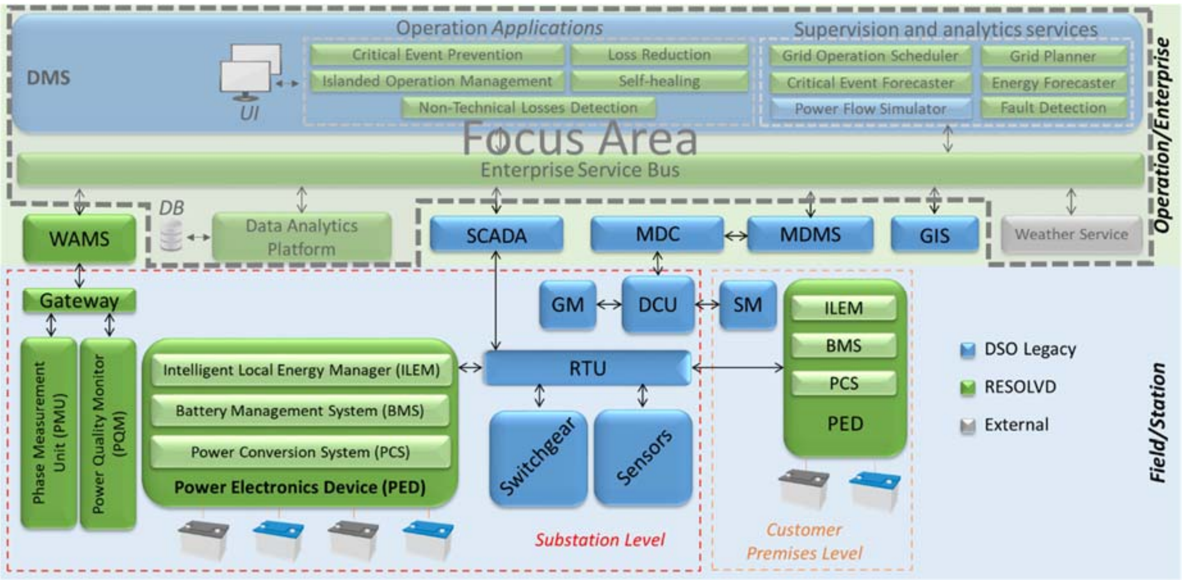

Figure 1: RESOLVD Solution & Focus Area

The artefacts of the project (in green colour) and their relation to the work reported in this document are described in the Table below. The area within the grey framework in Figure 1 indicates the focus area of the deployment testing work. The validation also includes interaction with components that are not part of this framework, but which have a direct link with ones inside it.

Operation Applications	Applications for managing advanced grid operations (i.e. Critical Event Prevention, Non-Technical Losses Detection, Losses Reduction, Self-Healing, Islanded Operation Management) and for providing a user interface for the human operator at the DSO's control centre.
Supervision and Analytics service	Enables advanced functionalities of the RESOLVD solution and supports Operation Applications, via providing energy and critical event forecasting, grid operation scheduling and fault detection.
Enterprise Service Bus	A middleware solution (ESB) that sits in between and integrates systems of the DSO's control centre (e.g. SCADA, MDMS), external systems (e.g. weather service) and RESOLVD artefacts. Facilitates information exchanges through the provision of standardised interfaces, service mediation and orchestration, message transformation as well as other functionalities that allow seamless and secure integration.
Data Analytics Platform	A central data repository and data analysis solution. Incorporates the following components: 1. A Triple store (i.e. Apache Jena[1]) for storing and querying grid data following the Common Information Model (CIM) [3] format (project custom profiles and CGMES [4] profiles) in their native form;

---

[1] https://jena.apache.org

	2. A NoSQL solution (i.e. Apache Cassandra[2]) for storing heterogeneous time series data; 3. Data processing tools (i.e. Apache Spark[3]) for handling complex queries
Power Electronic Device	A flexible grid asset enabling grid control actions (e.g. battery dispatch) as well as monitoring the grid at its connection point. It is composed of an Intelligent Local Energy Manager (ILEM), a Power Conversion System (PCS), a Battery Management System (BMS) and the batteries themselves. It is integrated to the RESOLVD backend through the SCADA.
Wide Area Monitoring System	System in charge of managing (collecting, concentrating, transmitting and monitoring) georeferenced field data coming from distributed sensors. Provides access to data of PMU and PQM devices and to the results of fault detection algorithms.

The RESOLVD solution is integrated with legacy systems of the DSO (e.g. SCADA, GIS, MDMS) and external weather services (e.g. Solcast – solcast.com, Dark SKY- darksky.net).

---

[2] http://cassandra.apache.org

[3] https://spark.apache.org

3. Deployment testing process

3.1. Methodology

Starting from the individual component level and gradually increasing in scope, the main objective of software testing is to identify defects and potential risks in the system and ensure that the end-result meets the business and user requirements of the project. A testing regimen should follow established testing pattern paradigms. A usual paradigm contains different layers of software testing, as indicated in the Table below.

Initial testing of individual components	Component/Unit testing focuses on testing the building blocks of an application, typically in isolation from other applications or units of the same application. A building block can be defined as a single class or method of a program that behaves as an atomic element. The developer of each solution has already performed such testing and it is outside of the scope of this report.
Testing of the different components and software modules working together	System integration testing focuses on the interactions and interfaces between systems, services, applications and their respective integrated software components. Integration testing concentrates on testing cross- building block communication, ensuring adherence to the agreed specifications of interfaces, data structures and communication protocols. In cases where there are dependencies between multiple blocks, testing can be incrementally performed for subsets, while gradually expanding to achieve full system testing.
End-to-end testing of the whole system, simulating real-world conditions and use cases	Functional testing focuses on verifying the behaviour and capabilities of a whole system, often considering the end-to-end tasks the system can perform. The main focus is on implementing detailed real-world test cases that test the end-to-end behaviour of the system as a whole. The test cases should be diverse and cover all individual system modules and possible real-word conditions and use cases. Additionally to testing the system under nominal operation (positive testing), the test cases should also ensure the system's capacity to handle and recover from adverse operational circumstances (negative testing).

Towards testing the deployed RESOLVD software solution, a test strategy was devised, documenting the test items, test approach to be used and validation means. In order to successfully plan the test process and achieve good test coverage of the underlying functional cases the following tasks were performed:

- Creation of a test plan for managing the test items, their risks and the tests' scheduling;
- Identification and documentation of main test cases, detailing the execution of tests;
- Identification and documentation of exceptional test cases, investigating possible points of failure in the application workflow to simulate adverse system conditions (e.g. failure of a system component, improper user behavior, unavailability of measurement data);
- Creation of test datasets containing input data required by test case/each module and capable of simulating a diverse set of test scenarios in near real-world conditions;
- Logging and reporting the detected issues towards identification of mitigation actions.

3.2. Items to be tested

The next table presents the list of RESOLVD software components that were deployed and tested, elaborating the features that were tested for each of them.

Module	***Features to be tested***
Meter Data Management System (MDMS)	• Collect energy metering data (energy consumption and generation) from smart metering devices and relay to ESB in intraday basis
Supervisory Control and Data Acquisition system (SCADA)	• Collect field data related to: o Switchgear configuration status o State of charge of batteries in the grid o Operational mode of the PED • Communicate with ESB module and relay meter data entries via SOAP/XML protocol • Relay scheduled battery set-points to the PED device for dispatch • Cancellation of battery schedule dispatch
Wide Area Monitoring System (WAMS)	• Communicate via MQTT protocol with the ESB and relay PMU/PQM measurement data entries in proximity to identified faults
Enterprise Service Bus (ESB)	• Provide authentication and authorization services • Act as a data broker and perform the necessary transformations to exchange data objects between DAP (persistence layer) and the rest of the platform modules. • Integrate with a third party API for retrieving temperature and irradiance forecast data (48 hours ahead) • Perform orchestration of the following workflows: o Critical event prevention o Losses reduction o Self-healing o Island power management
Data Analytics Platform (DAP)	• Communicate with the ESB for storage/retrieval of the following data: o SM measurements o Weather forecasts o Energy forecasts o Critical events o Power flow simulations o Battery status o Switch configuration o PED operation mode o Grid operation schedules o PMU/PQM data o Fault detection analysis o Grid models (EQ and GL CGMES profile datasets) • Pre-process and validate ingested data • Provide analytics services and calculate relevant KPIs (quality of provided measurements, availability of requirement measurements, grid operation performance KPIs)

Energy Forecaster (EF)	• (Re)train the forecasting algorithm using energy consumption and weather historical data from the ESB • Calculate day-ahead energy forecast using energy consumption, weather forecast and switch configuration data from the ESB • Communicate with the ESB module and relay energy forecast results via HTTPS/JSON protocol
Critical Event Forecaster (CEF)	• Calculate critical event forecasts using, energy demand/supply forecast and grid configuration data from the ESB • Communicate with the ESB module and relay critical event forecast results via HTTPS/JSON protocol
Grid Operation Scheduler (GOS)	• Perform analysis and generate the optimal grid schedules using energy demand/supply forecast and grid configuration data from the ESB • Communicate with the ESB module and relay grid-scheduling, power flow simulation results and performance metrics via HTTPS/JSON protocol
Fault Detection Application (FDA)	• Perform analysis of real-time PMU/PQM measurement data and detect, classify and localize faults occurring in the grid • Communicate with the ESB module and relay fault detection results via HTTPS/JSON protocol
Operation Applications	• Communicate with the ESB and request data required to render UI platform visualization via HTTP/JSON protocol • Provide a fluid UI and create visualizations of: o Grid model assets on map view o Domain events (e.g. forecasted critical events, detected faults) o Grid operation schedule (planned/dispatched) o Grid asset status (measured, forecasted) o Performance metrics (power loss reduction, islanding situation)

3.3. Testing environment

A testing environment is a setup of software and hardware components that enable the testing team to execute test cases. In other words, the testing environment supports the test execution process by providing a testbed comprising of all the required hardware, software and network infrastructure.

In the context of this work, the testing environment was the actual production environment enabling the realisation of communication with most of the real-life data flows, with the exception of that of WAMS, where a mock dataset was utilized to simulate the real behaviour of the system. A set of integration and functional tests were utilized to check that the system behaves as specified in the requirements.

The testing environment is presented in Figure 2. This environment ensured the availability of the necessary hardware and software that will be eventually validated during the piloting phase.

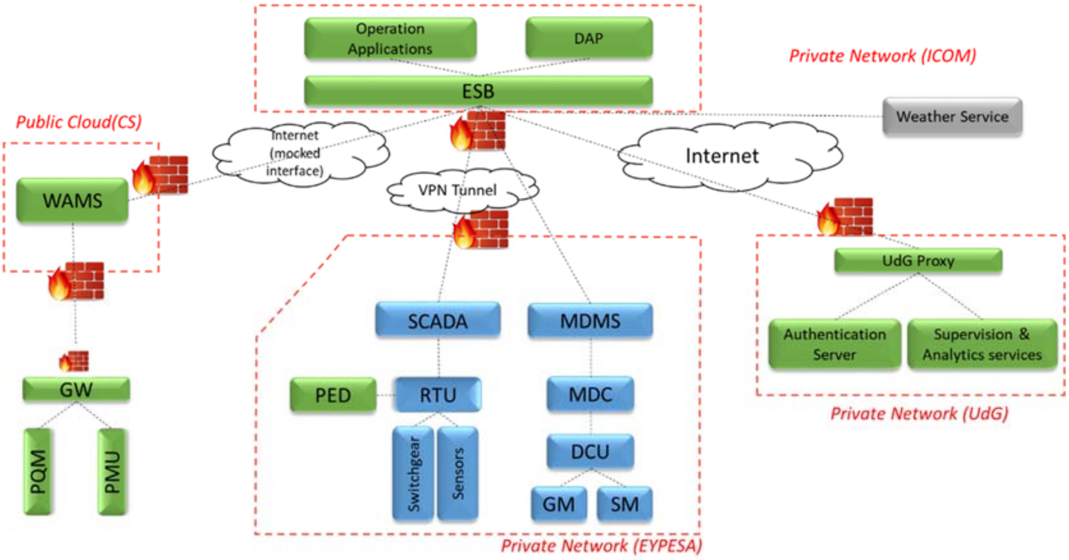

Figure 2 Testing environment

3.4. Test cases summary

This section provides a summary of the test cases that were defined to validate the deployment of the RESOLVD software solution in the production environment. Chapter 4 provides a detailed description of the cases detailing the modules participating in the test and their interactions and dependencies. Table 1 presents the list of test cases detailing the name, a short description, the participating modules, the test area and the dependency to other test cases.

The following test areas were devised to cover the area or interest:

1. **Accessing field sensing and external data**: The focus of these test cases is on data retrieval/receipt from various field' sensing devices as well as from the weather services.
2. **Integrating analytic services**: Concerns the integration of the various analytics modules. These test cases have a dependency on the test cases of area 1 as well as on test cases of the same level.
3. **Dispatching grid actuation commands**: Concerns the dispatch of command to the field devices. These cases have a dependency from test cases of area 1 and area 2.
4. **Operating the system through the user interface**: Tackles the end-user viewpoint, enabling the end-to-end testing of the solution, since the user interface integrates information for the execution of the whole business workflow

#	*Test Areas*	*Name*	*Description*	*Modules Participating*	*Testing Precursors*
1	***Accessing field sensing and external data***	Day-ahead weather data forecast request	In this scenario, the Weather Service (WS) module of the ESB periodically retrieves weather forecast data (temperature and solar irradiance) from external Web API's and relays them to DAP to be stored to the No-SQL database.	ESB(WS), DAP (Cassandra DB)	-
2		Smart meter data upload	In this scenario, the MDMS periodically relays SM data (energy consumption and generation) to the ESB in order to be stored in the database.	MDMS, ESB, DAP (Cassandra DB)	-
3		Switch status data retrieval	In this scenario, the ESB system periodically polls the SCADA service for the switchgear status. The data are then relayed to DAP in order be stored in the No-SQL database	SCADA, ESB, DAP (Cassandra DB)	-
4		Battery status data request	In this scenario the ESB system periodically polls the SCADA service for battery status data (state of charge and stored energy), the data are then relayed to DAP in order be stored in the No-SQL database.	SCADA, ESB, DAP (Cassandra DB)	-
5		PED operation mode request	In this scenario the ESB periodically polls the SCADA service to retrieve the operation mode of the ILEM (local or scheduled mode). The data are then relayed to DAP in order be stored in the No-SQL database.	SCADA, ESB, DAP (Cassandra DB)	-
6		Upload/Update of grid equipment model request	In this scenario, the grid equipment model (CGMES EQ profile file) is relayed to the ESB module to be stored in the database.	ESB, DAP (Jena Triplestore DB)	-

7		Upload/Update of grid geolocation model request (CGMES /GL)	In this scenario, the grid geolocation model (CGMES GL profile file) is relayed to the ESB module to be stored in the database.	ESB, DAP (Jena Triplestore DB)	-
8		PMU data upload request	In this scenario, WAMS relays real time PMU data to the ESB module in order to be stored in the No-SQL database. WAMS does not provide a continuous data feed, but only in proximity of a suspected fault.	WAMS, ESB, DAP (Cassandra DB)	-
9		Authorization token request	In this scenario, the ESB requests an authorization token for access to the analytics services.	ESB, UdG Authentication service	-
10	***Integrating analytic services***	Day-ahead forecast model (re)train request	In this scenario, the EF (re)trains the forecasting algorithm using historical energy consumption and weather data for a specific grid model.	ESB, DAP (Jena Triplestore DB)	1,2,9
11		Day-ahead forecast request	In this scenario, the EF receives weather forecast data and recent smart meter consumption and generation data and returns a day-ahead energy forecast for each node of the grid.	ESB, DAP, EF	1,2,9,10
12		Critical event forecast request	In this scenario the CEF receives energy demand/supply forecast data and switch configuration data and provides a critical event forecast for a specific time period, concerning a certain grid area.	CEF, ESB, DAP	3,9,11
13		Grid operation scheduling request	In this scenario, the GOS receives energy demand/supply forecast, and grid switch configuration and battery status data and provides an optimized grid schedule and power flow simulation data for a specific time period.	GOS, ESB, DAP	3,4,9,11

14		Fault detection analysis request	In this scenario, the FDA receives real time PMU data and switch status data and after analysing them provides ESB with information about the location, magnitude and type of detected grid faults.	FDA, ESB, DAP	3, 8, 9
15	***Dispatching grid actuation commands***	Battery set-point dispatch request	In this scenario, the ESB system relays calculated battery set-points for a selected time range to the SCADA service, in order to be dispatched by the PED.	SCADA, PED (ILEM), ESB, DAP	13
16		Battery set-point cancellation request	In this scenario, the ESB system relays a battery schedule cancellation signal to the SCADA service, in order to abort the execution of scheduled battery set-points already registered by the ILEM for dispatch.	SCADA, PED (ILEM), ESB, DAP (Cassandra DB)	15
17	***Operating the system through the user interface***	Visualization of grid model assets	In this scenario, the UI receives grid model data from the ESB and generates an interactive topological map of the grid assets.	Operation Applications, ESB, DAP (Jena Triplestore DB)	6,7
18		Visualization of dispatched grid configuration	In this scenario, the UI requests grid configuration data from the ESB and generates graphical representations of the statuses of the configurable assets of the grid (switches and batteries).	Operation Applications, ESB, DAP (Cassandra DB)	3,4
19		Visualization of scheduled grid configuration	In this scenario, the UI requests grid operation schedule data from the ESB and generates graphical representations of the scheduled statuses of the configurable assets of the grid (switches and batteries).	Operation Applications, ESB, DAP (Cassandra DB)	13,15

20		Visualization of forecasted grid status	In this scenario, the UI requests the outcome of the power flow analysis for future grid state from the ESB for specific grid assets and generates time series graphs of the forecasted voltages and currents.	Operation Applications, ESB, DAP (Cassandra DB)	13
21		Visualization of domain events	In this scenario, the operator is presented with the forecasted critical events, as retrieved by the CEF.	Operation Applications, ESB, DAP (Cassandra DB)	12, 15

Table 1 Test case summary

4. Test cases specification

This chapter provides a detailed presentation of the test cases that were designed to validate the deployment of the SW solution. The template utilized to document the test cases captures the description of the test case, the modules participating, stakeholders (partners) involved, preconditions and post conditions of the test, the detailed description of the sequence, input data as well as possible points of failure.

4.1. Accessing field sensing and external data

ITC1.	**Day-ahead weather data forecast request**
Description	In this scenario, the Weather Service (WS) module of ESB periodically retrieves weather forecast data (temperature and irradiance) from external Web API's and relays them to DAP to be stored to the No-SQL database.
Modules participating	ESB(WS), DAP (Cassandra DB)
Responsible parties	ICOM
Preconditions	o None
Expected results	o Day-ahead weather forecast data are successfully stored in the NO-SQL database every 1 hour.
Test Case Steps	1. WS periodically calls online weather API's (HTTPS GET) and receives temperature and solar irradiance forecast data, for a specific location. 2. WS transforms the data in proper format 3. WS relays the retrieved weather forecast data to DAP module (HTTP POST). 4. DAP stores the retrieved weather data to the No-SQL database.
Input data	o None
Possible Points of Failure	o Failure of communication between WS module and weather data API's. o Failure of communication between WS and DAP. o Incorrect data or missing data.

ITC2.	**Smart meter data upload**
Description	In this scenario, the MDMS periodically relays SM data (energy consumption and generation) to ESB in order to be stored in the database.
Modules participating	MDMS, ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA
Preconditions	o Collection of metering data by MDMS. o MDMS has a valid session for communicating with the ESB.
Expected results	o Meter measurement data (in one-hour granularity) are stored in the No-SQL database for available meters of the grid.
Test Case Steps	1. MDMS collects and aggregates meter measurement data from metering devices in the grid. 2. MDMS periodically relays meter data to ESB module (HTTPS POST). 3. ESB transforms the data in proper format 4. ESB forwards the data to DAP (HTTP POST) 5. DAP stores meter data to the No-SQL database.
Input data	o Valid meter data measurement feed in .csv format.
Possible Points of Failure	o Failure of communication between MDMS module and ESB module. o Failure of communication between ESB and DAP o Incorrect data, missing data, or incorrect data encoding.

ITC3. Switch status data retrieval

Description	In this scenario, the ESB system periodically polls the SCADA service for the switchgear status. The data are then relayed to DAP in order be stored in the No-SQL database
Modules participating	SCADA, ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA
Preconditions	o The SCADA monitors the switch statuses of all switches in the grid.
Expected results	o Switch status (15-minute time increments) data are stored in the No-SQL database.
Test Case Steps	1. The ESB polls the SCADA service (SOAP) for switch status data. 2. The SCADA service returns the switch status data (XML format). 3. ESB transforms the data in proper format. 4. ESB forwards the switch configuration data to DAP to be stored to the NO-SQL database.
Input data	o Switch configuration data.
Possible Points of Failure	o Failure of communication between SCADA module and DAP module. o Failure of communication between DAP and ESB. o Incorrect data, missing data, or incorrect data encoding.

ITC4. Battery status data request

Description	In this scenario the ESB system periodically polls the SCADA service for battery status data (state of charge and stored energy), the data are then relayed to DAP in order be stored in the No-SQL database.
Modules participating	SCADA, ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA
Preconditions	o The SCADA monitors the battery status.
Expected results	o Battery status data are stored in the No-SQL database for 15-minute time increments.
Test Case Steps	1. The ESB polls the SCADA service (SOAP) for battery status data. 2. The SCADA service returns the data (XML format). 3. ESB transforms the data in proper format. 4. ESB forwards the retrieved data to DAP, to be stored to the NO-SQL database.
Input data	o Battery status.
Possible Points of Failure	o Failure of communication between SCADA module and DAP module. o Failure of communication between DAP and ESB. o Incorrect data, missing data, or incorrect data encoding.

ITC5. PED operation mode request

Description	In this scenario the ESB periodically polls the SCADA service to retrieve the operation mode of the ILEM (local or scheduled mode), the data are then relayed to DAP in order be stored in the No-SQL database.
Modules participating	SCADA, ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA
Preconditions	o Monitoring of the PED mode of operation by the SCADA.

Expected results	o The PED mode of operation is stored in the No-SQL database in 15-minute time increments.
Test Case Steps	1. The ESB periodically polls the SCADA service (SOAP) to retrieve the PED mode. 2. ESB transforms the data in proper format. 3. ESB forwards the retrieved data to DAP, to be stored to the NO-SQL database.
Input data	o Valid data feed in XML format via SOAP messaging protocol.
Possible Points of Failure	o Failure of communication between ESB and SCADA. o Failure of communication between ESB and DAP. o Incorrect data, missing data, or incorrect data encoding.

ITC6. **Upload/Update of grid equipment model request**

Description	In this scenario, the grid equipment model (CGMES EQ profile file) is relayed to the ESB module to be stored in the database.
Modules participating	ESB, DAP (Jena Triplestore DB)
Responsible parties	ICOM
Preconditions	o Generation of a valid grid equipment model file that accurately describes the assets of the grid and their properties.
Expected results	o The grid equipment model file is successfully and accurately stored in the Triplestore database.
Test Case Steps	1. ESB accepts a valid grid equipment model file in CGMES EQ profile format (HTTPS POST). 2. ESB forwards the file to DAP (HTTP POST) 3. DAP processes the file and generates the equivalent grid asset entities. 4. DAP stores the grid model entities on the Triplestore database.
Input data	o Valid CIM grid equipment model file in the RDF/XML format.
Possible Points of Failure	o Failure of communication between ESB and DAP o Invalid CGMES EQ file.

ITC7. **Upload/Update of grid geolocation model request (CGMES/GL)**

Description	In this scenario, the grid geolocation model (CGMES GL profile file) is relayed to the ESB module to be stored in the database.
Modules participating	ESB, DAP (Jena Triplestore DB)
Responsible parties	ICOM
Preconditions	o Valid file that accurately describes the geographical locations of the assets of the grid.
Expected results	o Grid geolocation file is successfully stored in Triplestore database.
Test Case Steps	1. ESB accepts a valid grid location model file in CGMES GL profile format (HTTPS POST). 2. ESB forwards data to DAP (HTTP POST) 3. DAP processes the file and generates the equivalent entities 4. DAP stores the grid location model to the Triplestore database
Input data	o Valid CGMES grid geolocation model file in the RDF/XML format.
Possible Points of Failure	o Failure of communication between ESB and DAP. o Invalid CGMES GL file. o Mismatch of assets between EQ and GL files.

ITC8. **PMU data upload request**

Description	In this scenario WAMS relays real time PMU data to the ESB module in order to be stored in the No-SQL database. WAMS does not provide a continuous data feed, but only in proximity of a suspected fault.
Modules participating	WAMS, ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA, CS
Preconditions	o Collection and aggregation of valid PMU data by WAMS (100ms granularity). o Identification of potential faults through first pass analysis of collected data
Expected results	PMU data are stored in the No-SQL database in proximity to a suspected fault.
Test Case Steps	1. WAMS collects and aggregates data from various PMU devices situated throughout the grid. 2. WAMS performs a first stage statistical analysis of the PMU data and identifies abnormal values that may indicate a potential fault in the grid. 3. WAMS relays collected PMU data in proximity to the suspected fault to the ESB module (MQTT protocol). 4. ESB receives the data and forwards them data to DAP (HTTP POST) 5. DAP stores the retrieved PMU data to the NO-SQL database.
Input data	o Valid PMU data simulating faults and disturbances in the grid for a specific time period.
Possible Points of Failure	o Missing or invalid PMU data. o Failure of communication between WAMS and ESB.

4.2. Integrating analytic services

ITC9. **Authorization token request**

Description	In this scenario the ESB requests an authorization token for access to the analytics services.
Modules participating	ESB, UdG Authentication service
Responsible parties	ICOM, UdG
Preconditions	o Valid user credentials.
Expected results	o User is successfully authenticated and a valid authorization token is retrieved.
Test Case Steps	1. The ESB requests a token from UdG Authentication service, providing user credentials (HTTPS GET). 2. UdG Authentication service returns a valid authorization token back to ESB.
Input data	o User credentials
Possible Points of Failure	o Failure of communication between ESB and UdG authentication service. o Invalid user credentials.

ITC10. **Day-ahead forecast model (re)train request**

Description	In this scenario, the EF (re)trains the forecasting algorithm using historical energy consumption and weather data for a specific grid model.
Modules participating	ESB, DAP, EF

Responsible parties	ICOM, UdG
Preconditions	o Availability of historical weather data in DAP (**ITC1**). o Availability of historical consumption and generation smart meter data in DAP (**ITC2**). o Valid authorization token retrieved by ESB (**ITC9**).
Expected results	o Energy forecast model is successfully trained for all generation and consumption nodes that correspond to the ingested meter data.
Test Case Steps	1. The ESB retrieves historical consumption and weather data for the specific grid model from DAP (HTTP GET). 2. The ESB send a (re)train request to the EF, providing historical weather and smart meter data (HTTPS POST). 3. EF performs a (re)training of the energy forecasting models and returns a list of successfully trained models.
Input data	o Historic weather dataset for location corresponding to the grid (30+ days). o Historic meter reading dataset corresponding to the nodes of the specific grid (30+ days).
Possible Points of Failure	o Missing or invalid historical weather data. o Missing or invalid historical meter data. o Failure of communication between ESB and EF. o Failure of communication between ESB and DAP.

ITC11. Day-ahead forecast request

Description	In this scenario the EF receives weather forecast data and recent smart meter consumption and generation data and returns a day-ahead energy forecast for each node of the grid.
Modules participating	ESB, DAP, EF
Responsible parties	ICOM, UdG
Preconditions	o Availability of weather forecast data in DAP (**ITC1**) - 24 hours ahead. o Availability of historical consumption and generation MDMS data in DAP (**ITC2**) - last 170 hours o Valid authorization token retrieved by ESB (**ITC9**). o Availability of trained EF algorithm models for all nodes of the grid (**ITC10**).
Expected results	o EF returns a day-ahead energy forecast for all nodes of the grid for the next 24 hours.
Test Case Steps	1. The ESB retrieves historical consumption and weather data for a particular time interval and for a specific grid model from DAP (HTTP GET). 2. The ESB polls the EF for a generation and consumption forecast at a particular grid and for a particular time interval. (HTTPS POST) 3. EF performs a generation and consumption forecast at each node (or aggregation point) of the grid. 4. The EF responds to the ESB with the energy forecast for each node, for the next 24 hours (hourly granularity). 5. ESB forwards forecast data to DAP to be stored. (HTTP POST)
Input data	o Forecast time interval and grid model
Possible Points of Failure	o Missing or invalid weather forecast data. o Missing or invalid historical meter data. o Missing or incorrectly trained energy forecast models. o Failure of communication between ESB and EF. o Failure of communication between ESB and DAP.

ITC12. Critical event forecast request

Description	In this scenario the CEF receives energy demand/supply forecast data and switch configuration data and provides a critical event forecast for a specific time period, concerning a certain grid area.
Modules participating	CEF, ESB, DAP
Responsible parties	ICOM, UdG
Preconditions	o Availability of switch configuration data in DAP for the specific time interval (**ITC3**). o Valid authorization token retrieved by ESB (**ITC9**). o Availability of energy forecast data in DAP for the specific time interval (**ITC11**) - 24 hours ahead.
Expected results	o CEF returns forecasted critical events for the specific time period.
Test Case Steps	1. The ESB retrieves energy forecast data and switch configuration data for a particular time interval and for a specific grid model from DAP (HTTP GET). 2. ESB polls the CEF for a critical event forecast at a particular grid for a particular time interval. 3. CEF performs a critical event forecast for each node or line of the grid. 4. CEF responds to the ESB with the critical event forecast for the specific time interval (hourly granularity). 5. ESB forwards the CEF data to DAP to be stored (HTTP POST).
Input data	o Energy forecast dataset (24 hours-ahead) o Switch configuration data corresponding to the specific grid.
Possible Points of Failure	o Missing or invalid energy forecast data. o Missing or invalid switch configuration data. o Failure of communication between ESB and CEF. o Failure of communication between ESB and DAP.

ITC13. Grid operation scheduling request

Description	In this scenario, the GOS receives energy demand/supply forecast and grid configuration data and provides an optimized grid schedule and power flow simulation data for a specific time period.
Modules participating	GOS, ESB, DAP
Responsible parties	ICOM, UdG
Preconditions	o Availability of switch configuration data in DAP for the specific time interval (**ITC3**). o Availability of battery status data in DAP for the specific time interval (**ITC4**). o Valid authorization token retrieved by ESB (**ITC9**). o Availability of energy forecast data in DAP for the specific time interval (**ITC11**) - 24 hours ahead.
Expected results	GOS returns an optimized grid schedule and power flow simulation data for the specific time period.
Test Case Steps	1. The ESB retrieves energy forecast, switch configuration and battery status data for a particular time interval and for a specific grid model from DAP (HTTP GET). 2. ESB polls the GOS for a grid operation schedule (switch statuses and battery set points) at a particular grid for a particular time interval (HTTPS POST). 3. GOS performs a grid operation-scheduling forecast and selects the optimal grid configuration. 4. GOS responds to the ESB with the optimal grid configuration for the specific time interval (hourly granularity). 5. ESB forwards the GOS data to DAP to be stored. (HTTP POST)
Input data	o Energy forecast dataset (24 hours-ahead) o Switch configuration and battery status data corresponding to the specific grid.
Possible Points of Failure	o Missing or invalid energy forecast data. o Missing or invalid switch configuration data. o Missing or invalid battery status data. o Failure of communication between ESB and GOS. o Failure of communication between ESB and DAP.

ITC14. **Fault detection analysis request**

Description	In this scenario, the FDA receives real time PMU and switch status data and after analysing them provides ESB with information about the location, magnitude and type of detected grid faults.
Modules participating	FDA, ESB, DAP
Responsible parties	ICOM, UdG
Preconditions	o Availability of switch configuration data in DAP for the specific time interval (**ITC3**). o Availability of PMU data in DAP for the specific time interval (**ITC8**). o Valid authorization token retrieved by ESB (**ITC9**).
Expected results	Information about the duration, location, magnitude and type of detected grid faults are stored in the No-SQL database.
Test Case Steps	1. ESB receives PMU data batch from WAMS (MQTT protocol) 2. ESB retrieves switch configuration data from DAP (HTTP GET) 3. ESB relays the data to the FDA for fault detection analysis to be performed (HTTPS POST). 4. FDA analyses the data and returns information about the location, magnitude and type of detected grid faults. 5. ESB receives the FDA results and relays them to DAP to be stored.
Input data	o Valid PMU data batch in proximity to the suspected fault (100ms granularity)
Possible Points of Failure	o Missing or invalid switch configuration data. o Missing or invalid PMU data. o Failure of communication between ESB and FDA. o Failure of communication between ESB and DAP.

4.3. Dispatching grid actuation commands

ITC15. **Battery set-point dispatch request**

Description	In this scenario, the ESB system relays calculated battery set-points for a selected time range to the SCADA service in order to be dispatched by the PED.
Modules participating	SCADA, PED (ILEM), ESB, DAP
Responsible parties	ICOM, EYPESA, UPC
Preconditions	o Valid battery set point schedule (1 hour granularity) stored in the NO-SQL database for the selected time-range (**ITC13**).
Expected results	o Battery set-points are successfully relayed to the PED device for dispatch.
Test Case Steps	1. ESB requests the battery set-point schedule for a specific time period from DAP (24 hour interval). 2. DAP queries the NO-SQL database, retrieves the requested data and returns them to the ESB. 3. ESB transforms the data in proper format. 4. ESB forwards the data to the SCADA set-point service (SOAP/XML). 5. The SCADA service relays the schedule to the PED device for dispatch. 6. The PED dispatches the battery set-point schedule.
Input data	o Valid grid operation battery set-point schedule calculated for the selected time period (24 hour interval).
Possible Points of Failure	o Failure of communication between ESB and DAP. o Failure of communication between ESB and SCADA. o Failure of communication between SCADA and the PED device. o Failure of optimization algorithm within the PED o Incorrect or missing battery schedule data.

ITC16. Battery set-point cancellation request

Description	In this scenario, the ESB system relays a battery schedule cancellation signal to the SCADA service, in order to abort the execution of scheduled battery set-points already registered by the ILEM for dispatch.
Modules participating	SCADA, PED (ILEM), ESB, DAP (Cassandra DB)
Responsible parties	ICOM, EYPESA, UPC
Preconditions	o None
Expected results	o All scheduled battery set-points dispatched to the ILEM are cancelled (energy exchanged by the battery is set to 0).
Test Case Steps	1. ESB receives a schedule cancellation signal through the web interface. 2. ESB creates and sends a cancelation schedule request to the SCADA set-point service (SOAP/XML). 3. The SCADA service relays the schedule to the PED for dispatch. 4. The PED dispatches the battery set-point schedule for the specified period.
Input data	o Valid grid operation battery set-point schedule calculated for the selected time period (24 hour interval).
Possible Points of Failure	o Failure of communication between ESB and SCADA. o Failure of communication between SCADA and the PED. o Failure of communication between ILEM and PCS, within the PED

4.4. Operating the system through the user interface

ITC17. Visualization of grid model assets

Description	In this scenario, the UI receives grid model data from ESB and generates an interactive topological map of the grid assets.
Modules participating	Operation Applications, ESB, DAP (Triplestore DB)
Responsible parties	ICOM
Preconditions	o Valid grid equipment model (CGMES EQ profile file) uploaded in the Triplestore DB (**ITC6**). o Valid grid geolocation model (CGMES GL profile file) uploaded in the Triplestore DB (**ITC7**).
Expected results	The UI returns a valid model representation of all the grid assets and their properties, mapped on a geographical map that corresponds to the individual assets' actual locations.
Test Case Steps	1. The UI platform polls the ESB for grid model data corresponding to the particular grid (HTTP GET). 2. ESB forwards the request to DAP and DAP retrieves all relevant grid model data from the Triplestore DB. 3. DAP processes the data and creates the geoJSON objects corresponding to each individual grid asset. 4. DAP forwards the data to ESB which returns them to the UI platform for visualization. 5. The UI platform processes the data and generates a map view of the grid assets.
Input data	o Availability of valid grid equipment model data (CGMES EQ profile). o Availability of valid grid geolocation model data (CGMES GL profile).
Possible Points of Failure	o Missing or invalid grid equipment model data. o Missing or invalid grid geolocation model data o Failure of communication between Platform UI and ESB.

ITC18. Visualization of grid status

Description	In this scenario, the UI requests grid status data from the ESB and generates graphical representations of the statuses of the configurable assets of the grid (switches and batteries).
Modules participating	Operation Applications, ESB, DAP (Cassandra DB)
Responsible parties	ICOM
Preconditions	o Availability of grid status data (switch states and battery statuses provided from SCADA) in DAP, for the specific time interval (**ITC3, ITC4**).
Expected results	The UI returns a series of time series graphs that depict the status of each of the switch and battery assets of the grid, for the specific time period.
Test Case Steps	1. The UI platform polls the ESB for grid status data for the specific time interval (HTTP GET). 2. ESB forwards the request to DAP. 3. DAP retrieves the requested data for each of switch and battery assets of the grid, from the No-SQL database. 4. DAP forwards the data to ESB which returns them to the UI platform for visualization. 5. The UI platform processes the data and generates time series graphs of data for the selected time period (24 hour intervals).
Input data	o Battery status and switch configuration data for the current time period (hourly granularity).
Possible Points of Failure	o Missing or invalid battery status of switch configuration data. o Missing or invalid grid topology model. o Failure of communication between UI and ESB.

ITC19. Visualization of scheduled grid configuration

Description	In this scenario, the UI requests grid operation schedule data from the ESB and generates graphical representations of the dispatched schedules of the configurable assets of the grid (switches and batteries).
Modules participating	Operation Applications, ESB, DAP (Cassandra DB)
Responsible parties	ICOM
Preconditions	o Availability of grid operation schedule data in DAP for the specific time interval (**ITC13**). o Successful visualization of the grid model assets (**ITC15**).
Expected results	The UI returns a series of time series graphs that depict the scheduled status of each of the switch and battery assets of the grid, for a specific time period.
Test Case Steps	1. The UI platform polls the ESB for grid operation schedule data for the specific time interval (HTTP GET). 2. ESB forwards the request to DAP. 3. DAP retrieves the requested data for each of switch and battery assets of the grid, from the No-SQL database. 4. DAP forwards the data to ESB which returns them to the UI platform for visualization. 5. The UI platform processes the data and generates time series graphs of data for the selected time period (24-hour intervals).
Input data	o Grid operation schedule data (scheduled switch states and battery set-points) for the requested time interval.
Possible Points of Failure	o Missing or invalid grid operation schedule data. o Missing or invalid grid topology model. o Failure of communication between UI and ESB.

ITC20. Visualization of forecasted grid status

Description	In this scenario, the UI requests the outcome of the power flow analysis for future grid state from the ESB for specific grid assets and generates timeseries graphs of the forecasted voltages and currents.
Modules participating	Operation Applications, ESB, DAP (Cassandra DB)
Responsible parties	ICOM
Preconditions	o Availability of PFS data in DAP for the specific time interval and grid asset (**ITC13**).
Expected results	The UI returns a series of graphs that depending of the type of asset, either depict the forecasted voltage of the node (or aggregation point), or the forecasted current of the line (or branch).
Test Case Steps	1. The UI platform polls the ESB for PFS data for the specific time interval and grid asset (HTTP GET). 2. ESB forwards the request to DAP. 3. DAP retrieves the requested PFS data for each asset from the No-SQL database. 4. DAP forwards the data to ESB which returns them to the UI platform for visualization. 5. The UI platform processes the data and generates time series graphs of data for the selected time period (24-hour intervals).
Input data	o PFS data for the selected time interval and asset (hourly granularity).
Possible Points of Failure	o Missing or invalid PFS data. o Missing or invalid grid topology model. o Failure of communication between UI and ESB.

ITC21. Visualization of domain events

Description	In this scenario, the operator is presented with the forecasted critical events, as retrieved by the CEF.
Modules participating	Operation Applications, ESB, DAP, Cassandra DB
Responsible parties	ICOM
Preconditions	o Availability of Critical Event data in DAP for the specific time interval (**ITC10**). o Successful visualization of the grid model assets (**ITC15**).
Expected results	The UI returns a list of the forecasted critical events for the selected time period. The UI also maps the forecasted critical events to their respective assets in the grid model graphical map.
Test Case Steps	1. The UI platform polls the ESB for CEF data for the specific time interval (HTTP GET). 2. ESB forwards the request to DAP. 3. DAP retrieves the requested CEF data from the No-SQL database. 4. DAP forwards the data to ESB and ESB returns them to the UI platform for visualization. 5. The UI platform processes the data, maps them to individual assets and generates the visualizations.
Input data	o CEF data for the selected time interval (hourly granularity).
Possible Points of Failure	o Missing or invalid CEF data. o Missing or invalid grid topology model. o Failure of communication between UI and ESB.

5. Results

5.1. Test results

Test #	***Test Area***	***Scenario***	***Status***	***Notes***
ITC1	***Accessing field sensing and external data***	Day-ahead weather data forecast request	**Passed**	
ITC2		Smart meter data upload	**Passed***	*After meta-analysis of the data collected during the testing phase, issues with the consistency of the data provided by the MDMS module were discovered. More specifically, the data feed had significant quality issues, with multiple records missing between consecutive measurements, in some cases even approaching 30% of the total record count.* *To mitigate this problem the data feed was switched to a different data source, which could only report measurements about imported and exported power and not on voltage, but is sufficient for simulations conducted by the platform.* *This change caused a significant improvement on the quality of the collected datasets.*
ITC3		Switch status data retrieval	**Passed***	*Integration/communication among ESB and SCADA was conducted with success, but the actual switch status from the physical devices and the SCADA is still pending. Mocked data were used for the realization of interdependent tests.*
ITC4		Battery status data request	**Passed**	
ITC5		PED operation mode request	**Passed**	
ITC6		Upload/Update of grid equipment model request	**Passed**	
ITC7		Upload/Update of grid geolocation model request	**Passed**	
ITC8		PMU data upload request	**Passed***	*Testing was conducted using a simulated data feed from the WAMS. This enabled also to replicate the occurrence actual faults in the grid.*
ITC9	***Integrating analytic services***	Authorization token request	**Passed**	
ITC10		Day-ahead forecast model (re)train request	**Passed**	
ITC11		Day-ahead forecast request	**Passed**	
ITC12		Critical Event Forecast request	**Passed**	
ITC13		Grid operation scheduling request	**Passed**	

ITC14		Fault detection analysis request	**Passed***	*Testing was conducted using a simulated PMU dataset (see ITC8).*
ITC15	***Dispatching grid actuation commands***	Battery set-point dispatch request	**Passed**	
ITC16		Battery set-point cancellation request	**Passed**	
ITC17	***Operating the system through the user interface***	Visualization of grid model assets	**Passed**	
ITC18		Visualization of dispatched grid configuration	**Passed**	
ITC19		Visualization of scheduled grid configuration	**Passed**	
ITC20		Visualization of forecasted grid status	**Passed**	
ITC21		Visualization of domain events	**Passed**	

Table 2 Test results summary

5.2. Identified issues and mitigation actions

The software part of the RESOLVD solution has several data feeds: real power injections at loads and generators, monitored by the MDMS and transmitted to RESOLVD backend platform; switchgear statuses, monitored and transmitted over the SCADA to the RESOLVD backend platform (switch status might be unknown from the source); weather data, provided by external services. These feeds are utilized as input (direct or indirect) by various algorithms, e.g. by the energy forecasting, the critical events forecasting, and the optimal scheduling. During the deployment phase, data integrity issues were identified, which pose a high risk, as bad quality or unavailability of regular measurement data may produce errors or even completely disrupt the operation of these algorithms.

During the deployment phase data integrity issues were discovered, located mostly on the data feed of the MDMS. The original data feed had significant quality issues, with multiple records missing between consecutive measurements, in some cases even approaching 30% of the total record count. This unavailability of data had a significant impact on the energy forecasting operation and by extension to most of the analytics operations of RESOLVD solution.

To mitigate this problem the data feed was switched to a different data source (pre-process data from billing system), with lower granularity and fewer available monitored attributes (only energy exchanges). This shift caused a significant improvement on the quality of the collected datasets whilst the worse "quantity" of available attributes and granularity did not affect the proper operation of the analytics services.

Further mitigation actions that were identified to tackle data integrity issues, concern:

❑ Validation of the accuracy and completeness of incoming data, through a continuous monitoring process of the health of data sets

❑ Introduction of data pre-processing (e.g. outlier removal, missing data values estimation) for increasing the robustness of analytics computations

6. Conclusions and next steps

This report documented the deployment and testing of the software part of the RESOLVD solution in the validation environment, presenting methodology, test cases, results and lessons learnt. The identified issues concerned mostly data quality and availability, for which mitigation actions were defined for the different stakeholders. From the perspective of the analytics provision, a more robust design was devised for tackling data issues; from the perspective of integration, a process for continuous monitoring of the health of incoming data was implemented; from the perspective of the end user and data provider, the importance of selecting the appropriate data sources for such services was highlighted.

The work documented in this report, along with the documented work of *deliverable D5.2 "Report on HW installation with feedback to activities related to grid actuation, platform integration and interoperability with legacy systems"* [5], aim to set the grounds for the realisation of the pilot, which will take place in the next phase of the project.

References

[1] RESOLVD, "D1.3 – Interoperability and Integration Analysis and Requirements".

[2] RESOLVD, "D4.1 – Detailed description of the platform".

[3] "https://en.wikipedia.org/wiki/Common_Information_Model_(electricity)," [Online].

[4] "Common Grid Model Exchange Standard (CGMES) Library.," [Online]. Available: https://www.entsoe.eu/digital/cim/cim-for-grid-models-exchange/.

[5] RESOLVD, "D5.2 Report on HW installation with feedback to activities related to grid actuation, platform integration and interoperability with legacy systems".