MARKET ENABLING INTERFACE TO UNLOCK FLEXIBILITY SOLUTIONS FOR COST-EFFECTIVE MANAGEMENT OF SMARTER DISTRIBUTION GRIDS

Deliverable: D8.2

German Demonstrator — Demonstration of congestion management using market-based flexibility in the LV grid

Specifications of test scenarios within the German Demonstrator

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864334

H2020 – LC-ES-1-2019

Document	
D8.2	Specifications of test scenarios within the German Demonstrator

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

Author(s)	Institution	Contact (e-mail, phone)
Gesa Milzer	NODES	gesa.milzer@NODESmarket.com
Paul Kratsch	E.ON	paul.kratsch@eon.com
Helene Ask Uggla	E.ON	helene.ask-uggla@eon.se
David Brummund	MNS	david.brummund@mitnetz-strom.de
Kseniia Sinitsyna	MNS	kseniia.sinitsyna@mitnetz-strom.de
Mahtab Kaffash	Centrica	mahtab.kaffash@centrica.com
Evelyn Heylen	Centrica	evelyn.heylen@centrica.com
Pierre Crucifix	N-SIDE	pcu@n-side.com
Louise Adam	N-SIDE	lad@n-side.com
Reinhilde D'hulst	VITO	Reinhilde.dhulst@vito.be
Md Umar Hashmi	KU Leuven	mdumar.hashmi@kuleuven.be
Simon Nagels	KU Leuven	simon.nagels@kuleuven.be
Gil Silva Sampaio	INESC TEC	gil.s.sampaio@inesctec.pt

Key word	Demonstration
Due Delivery Date	2023/06/30
Date of Delivery	2023/06/28

Document version	Date	Change
D8.2_V1	2023/05/31	1st draft for internal review
D8.2_V2	2023/06/08	First complete draft for official review
D8.2_V3	2022/06/28	Final version for submission

Reviewers (Institution, Name)		Email	Validation date
E-REDES	Rita Mourão	ritalopes.mourao @e-redes.pt	2023/06/26
MIKRO	Michał Konopiński	michal.konopinski@mikronika.com.pl	2023/06/26

Table of Contents

List of Abbreviations

BEMS	Business Energy Management System
BUC	Business Use Case
CC	Chance Constraint
D	Deliverable
DdSE	Data driven State Estimation
DN	Distribution Network
DNI	Distribution Network Incidents
DSO(s)	Distribution System Operator
EV	Electric Vehicles
FNA	Flexibility Needs Assessment
HEMS	Home Energy Management System
KPI	Key Performance Indicator
LV	Low Voltage
MNS	Mitnetz Strom
MV	Medium Voltage
MVP	Minimum Valuable Product
OBR	Optimal Bid Recommender
OPF	Optimal Power Flow
PV	Photovoltaic
REST API	RESTful application programming interface
SO(s)	System Operator
SSH	Secure Shell
SUC	System Use Case
T	Task
UMEI	Universal Market Enabling Interface
WP(s)	Work Package

List of Tables

List of Figures

Executive Summary

This Deliverable has been drafted in the context of the EUniversal project. The project aims to overcome existing limitations regarding the use of flexibility by DSOs for congestion and grid management. Considering the European approach as well as the need for harmonization and creation of standards, one objective of EUniversal is the establishment and integration of the Universal Market Enabling Interface (UMEI) to ensure system interoperability to facilitate access to multiple flexibility market platforms and thus access to distributed flexibility. The UMEI is tested in three locations across Europe, i.e. Portugal, Germany and Poland, examining its use for market-based flexibility procurement in various use cases.

This report follows up on the deliverable D8.1 for the German Demonstrator "Specifications and guidelines of tools for an Active LV grid for field testing". The deliverable describes the technical and operational details of the site-specific problem to solve, the pilot site, each tool implemented in the digital flexibility value chain and the interconnection between consecutive tools. Furthermore, D8.1 outlines the specific use of the UMEI to facilitate access to flexibility services via a market-based approach.

This deliverable reports the set up and test scenarios of the flexibility value chain within the German demonstrator (WP8) starting from congestion detection to the market-based flexibility service procurement. The tests are divided into two parts: 1) Individual tests of each smart grid tool, the market environment and associated functions of the DSO, FSP and the Optimal Bid Recommender to ensure correct functioning and information exchange and 2) the operational testing of consecutive members of the digital flexibility value chain. Due to time constraints as well as operational challenges, this report outlines the test part 1), related findings and challenges of the performed test series. Wherever possible, the functionalities and operational processes will be measured against the Key Performance Indicators (KPIs) identified in WP2. These results will be presented in detail in Deliverable 8.3 together with the results of the test series 2).

1. Introduction

1.1 Background

The European Union aims at transforming the energy system towards a sustainable, low-carbon and climate-friendly economy. The scope is to increase the energy share of electricity production in distribution grids to around 50% of renewable energy sources (RES) until 2030 while guaranteeing the security of supply and avoiding unnecessary network investments. For this purpose, load generation and consumption of prosumers across all grid levels shall serve as energy and flexibility resources making them active participants in the energy system. In such a scenario, prosumers become key enablers towards a more sustainable, low-carbon and climate-friendly electricity system by adapting their consumption and production behaviour to stabilize the grid when needed. Yet, flexibility will also add complexity and create unpredictable power flows in the distribution networks. Distribution System Operators (DSOs) need to integrate smart-grid solutions to cope with the new types of load patterns of diverse small-scale assets (e.g. electric vehicles and heat pumps) and to identify the required flexibility to safely host the increasing share of RES. Therefore, innovative technologies and solutions are required to transform the challenges of the energy transition into opportunities for the sector, and ultimately for the society.

The EUniversal project aims to overcome the existing challenges for DSOs concerning the use of flexibility. The primary project goal is to overcome barriers between multiple market agents and their internal systems through the Universal Market Enabling Interface (UMEI) (D2.4-D2.6). The UMEI has been developed to support distribution system operators and their active system management by facilitating access to distributed flexibility via multiple market platforms at different locations while limiting the DSO system changes to a minimum. The UMEI is tested in three different demonstrations in Germany, Poland and Portugal. This deliverable describes the technical specifications and test scenarios performed in the German demonstrator.

The German demonstrator led by German DSO Mitnetz Strom examines the operational and functional viability of each element required within the digital flexibility value chain, starting from smart grid tools to identify existing and future congestions in terms of location, volume and direction, to using an optimal bid recommender to select the optimal bid available on the local flexibility market (Figure 1.1).

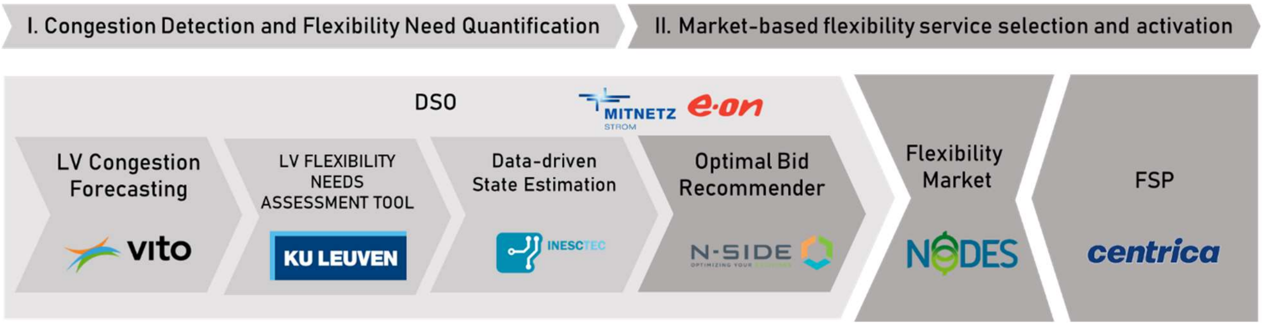


Figure 1.1- Simplified Overview of the smart grid tools and the market environment

1.2 Applicability and limitations

The preparation and realization of the field test encounters numerous technical, operational, and regulatory challenges, pointing out existing barriers that impede the successful use of distributed flexibility for grid management:

Technical challenges

Technical applications and challenges are mainly related to the integration and interconnection of the different tools and APIs (mainly the UMEI and associated functional specifications) within the different systems. The smart grid tools developed within EUniversal and the underlying algorithms must be correctly integrated into the German DSO Mitnetz Strom (MNS) system to ensure the well-functioning of each component within the entire value chain, and to achieve sufficient accuracy and precision of the results. Furthermore, the use of flexibility requires intelligent Home Energy Management Systems (HEMS) and Business Energy Management Systems (BEMS) to provide a control mechanism of flexibility activation and data monitoring for validation of flexibility delivery. The test preparations have shown the enhanced need for standardization between technical devices for energy monitoring as well as the installation in place to ensure correct and effective functioning.

Another technical limitation concerns the management of active and reactive power, BUC I and BUC II in the German Demonstrator, respectively. The clear separation of both may not be realizable from a grid perspective. Considering the market-based procurement at this point, trading of reactive power as a flexibility product is still quite inaccurate due to limitations in quantification of the required volume and pricing. The presented tests within the German demonstrator therefore only concern active power to overcome technical and operational barriers in a first place.

Operational challenges

From an operational perspective, the lack of active user participation, as well as meter data and grid data in terms of amount, type, and quality of data, as well as the required granularity, complicates the reliable replication of a realistic grid problem and solution. The input of this data directly affects the accuracy of the output of the smart grid tools and hence their applicability and reliability for the DSO. Sufficient data is crucial for the Demonstrator to reliably evaluate the cost-effectiveness of the flexibility usage in the test locations.

To test the set-up as well as the entire value chain despite of partly insufficient data, a series of simulations has been performed. However, it is important to note that a simulation will only serve as an operational test scenario but does not reliably replicate a realistic situation.

The technical and operational challenges, however, persist to a large extent due to the underlying regulatory framework in Germany concerning an effective Smart Meter Rollout, grid tariff structures, taxes as well as mandatory and voluntary redispatch measures to support the smart use of available distributed flexibility for the purpose of grid management.

Regulatory challenges

Germany has implemented two methods of steering flexibility, one being the mandatory Redispatch 2.0 and the other one being cost-based. The latter is mainly regulated through the paragraph §14a in the German Energy Industry Act (EnWG). The regulators are currently working on an updated version to amplify the roles and resources to use the distributed flexibility while maintaining the existing redispatch measures to a large extent.

On one hand, the use of market-based flexibility shall reduce or even prevent additional costs and bridge time delays of the grid infrastructure expansion. On another, the market-based flexibility helps to use the available resources efficiently and effectively by solving grid problems locally through smart regulation of local or regional assets. Besides, implementing flexibility markets to benefit from the available distributed flexibility for grid services requires an adaption of the German regulatory framework as well as adjustments of the grid tariff and tax schemes to incentivize the participation of flexibility providers. A detailed evaluation of the required adaptions has been elaborated by a forum of experts and presented to the German Ministry for Economic Affairs and Climate Action in June 2021. This adaptation has, however, not yet led to an alteration in the regulatory framework, and in the German Easter package (2022), market flexibility was only mentioned as an alternative measure with the flexibility potential being categorized as undetermined benefits. This categorization is especially important considering the challenges and risks of integrating flexibility markets, such as the increasing need for coordination between system operators as well as the coordination of numerous new assets and asset types that contribute to the load flow. Furthermore, especially in Germany the potential risk of strategic bidding to artificially increase the revenue of market participants is a major argument against the market-based approach.

With the ongoing work on the German Demo, the goal is to showcase the digital value chain of flexibility suggesting innovative solutions to overcome the existing technical and operational challenges.

1.3 Report structure

In the following chapters a summary of the pilot area and selected grid zones is presented together with the identified BUC and SUCs from WP2. In chapter 3 the set-up and integration of each tool into the DSO system to ensure correct functioning and interoperability is explained. Chapter 4 includes the performed test series and results per tool and market participant to prove technical and operational feasibility of each tool and component. The last chapter includes the conclusion of the previously described tests followed by a short overview of the next steps.

This report is using valuable information from other WPs, namely:

- WP2, for the definition of use cases that will be demonstrated, as well as the UMEI API functional specification, namely with the identification of the interactions between the DSO and Flexibility Market platforms and data exchange
- WP3, with the use of a flexibility toolbox, identifying the technologies and solutions most suitable to provide flexibility services to the distribution grid
- WP4, for the development of the DSO smart grid tools and their alignment
- WP5, the identification of relevant market mechanisms
- WP6, with a common framework to harmonise, monitor and assess the validation of the result

The results of the pilot simulations and measures will be presented and explained in D8.3. The knowledge gained in this demonstration is also used to support WP10 in the development of business models for the exploitation of EUniversal's results and to provide recommendations for policy makers and regulatory authorities to set up a framework for flexibility markets.

2 General overview of the German demonstrator

2.1 Demo site characteristics

The German demonstration is conducted in LV Grids of the German DSO Mitnetz Strom (MNS - E.ON linked third party). The supply area of MNS is in the Eastern Germany, and includes parts of Brandenburg, Saxony-Anhalt, Saxony and Thuringia. The region is home to 2.3 million inhabitants in an area of 30,804 km^2. The length of the grid is about 6,000 km in HV, about 24,000 km in MV and about 44,000 km in LV Level. MNS operates and maintains about 17,000 substations with an installed capacity of more than 5,000 MVA. Therefore, the grid provides a broad variety of scenarios and can demonstrate the use cases defined in WP2, ie. Congestion Management & Voltage Control with market-based active/reactive power flexibility (2.3.1 Business Use Cases (BUC).

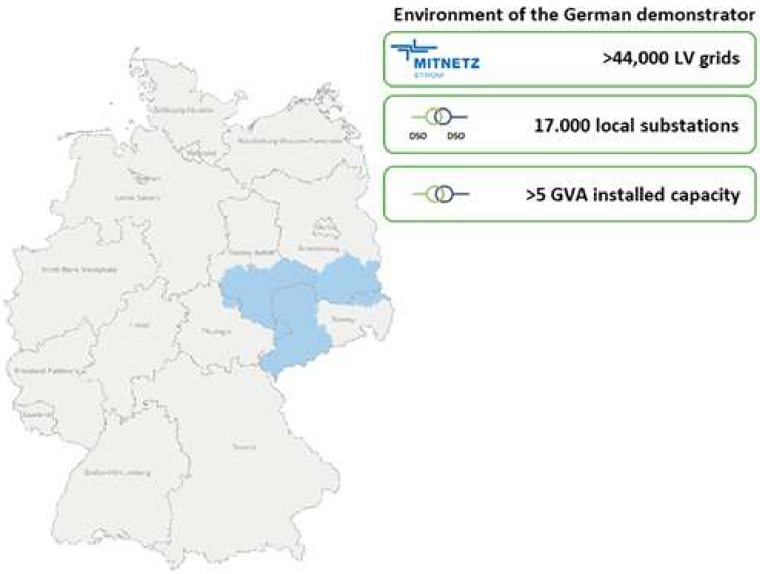


Figure 2.1- Demo site characteristics

To achieve the objectives of the demonstration, adequate grid sections with a relevant infeed of RES and cross-sector loads (e.g. storage and heat pump users) were selected considering grid size and topology, past congestion/voltage problems and the number and type of flexible devices in the network.

2.2 Grid selection

The grid areas for the German were selected considering grid size and topology, past congestion/voltage problems and aiming to embrace the largest possible number and variety of flexible applications to enable different scenarios for the field to replicate the future supply task of the distribution grid operators. The selected grids are among the MNS low-voltage grids with the highest densities of flexible resources and are representative of the possible usage of flexibility markets at the low-voltage level. In addition, a further increase of steerable systems is expected for the coming years, which makes the selected regions an optimal use case for grid congestion solutions. In a second step two pilot regions were selected in the MNS grid regions of Brandenburg and West Saxony. However, since customer acquisition was only very moderately successful (chapter 3.3), it was decided to concentrate on the LV grids with larger cable expansions and set-up measuring support points and to use them for the evaluation and prognosis of the grid state.

***Table 2.1 - LV grid areas of the German Demonstrator*[1]**

Identifier	MLq0094	MFn4420
Grid region	Brandenburg	West-Saxony
Town where the LV Grid is located	Falkenberg\Elster	Brandis
Number of connected meters[2]	400	300
Number of customers with flexible devices	50	44
Specifics	• Mostly single-family houses • Partly still use of night storage heaters (historically grown) • MNS site in town	• Residential area with single-family houses from the 1990s/2000s • Radial grid structure

2.3 Mapping of Use Cases and Key Performance Indicators

Starting from the flexibility needs from DSOs, explored, and defined in D2.1 of the project, a set of Business Use Cases (BUCs) were determined for each demonstrator in the project, each describing the interactions needed between the involved stakeholders and their associated information requirements. This description can be found in D2.2.

1 Rounded values

Following these BUCs, a set of System Use Cases (SUCs) was defined, identifying clear functional specifications of the operational prototypes in the project test environments. This description can be found in D2.3.

In a next step, a set of key performance indicators (KPIs) was defined to evaluate and monitor the performance of the project tests and pilots. These KPIs are based on the previously defined BUCs and SUCs . These KPIs and the detailed process on how they are defined can be found in deliverable D6.3.

The following section provides an overview of the BUCs, SUCs and KPIs that are relevant to the German demonstrator.

2.3.1 Business Use Cases (BUC)

A BUC describes the steps and activities in a process that are necessary to achieve a business goal. Two BUCs were identified to be of predominant importance for the German Demonstrator:

Table 2.2- Business Use Cases of the German Demonstrator

ID	Name	Service	Mechanism	Main Steps
DE-AP	Congestion Management & Voltage Control with market-based active power flexibility	Congestion management and Voltage control	Local flexibility market	• Identification and localisation of required flexibility (smart grid tools) • Registration and Prequalification assets • Bidding and selection • Delivery and monitoring (not tested) • Settlement (not tested)
DE-RP	Congestion Management & Voltage Control with market-based reactive power flexibility	Voltage control and congestion management	Local flexibility market	• Identification and localisation of required flexibility (smart grid tools) • Registration and Prequalification • Bidding and selection • Delivery and monitoring (not tested) • Settlement (not tested)

Although the two BUCs, Congestion Management & Voltage Control with market-based active and reactive power flexibility, are clearly separated in Table 2.2, such a clear distinction may not be realizable or useful from a grid perspective. The converters, especially in LV, are generally designed to have an operating range that is fully or partially dependent on active power. Furthermore, LV grids are resistive and the separation from a technical perspective is difficult. Therefore, the algorithms of the smart grid tools do not distinguish between active and reactive power as interdependent components.

The applied market clearing algorithms consider only one parameter that can be either active or reactive power. Both can be traded as separate products on the market platform assuming a correct quantification of the required dimensions, pricing and associated baselines. Nevertheless, the separation of both as a product may cause a trade-off and affect the efficiency of the market.

Therefore, within WP8.2, during the testing and pilot phase, the focus lies on BUC I DE-AP and related SUCs as a first step.

2.3.2 System Use Cases (SUC) of the German Demonstrator

The SUCs, in contrast to the BUCs, provide detailed descriptions of the process itself. The SUCs give the functional description needed to support the BUCs by detailing which activities are performed, who is going to execute them and on which system.

In the German Demonstrator, 10 System Use Cases have been defined within Task 2.3 and were classified into three domains:

- Smart Grid Operations – including all use cases involving the distribution network operation and planning
- Flexibility Market – including all use cases involving local flexibility market operation
- Flexibility Aggregation and grid users – including all use cases related to the FSP and the consumer/active customer provision of flexibility services in the market.

The list of System Use Cases defined for the German demonstration is presented in Table 2.3.

Table 2.3 - System Use Cases of the German Demonstrator

Domain	SUC ID	SUC name	BUC ID	Related Pilot(s)	Owner
Smart Grid Operation	SUC 4	Day-ahead congestion management considering flexibility needs in LV and MV networks	DE AP, DE RP	PT, DE	INESC TEC
	SUC 5	Estimating LV voltage magnitude based on historical data and load forecasts	DE AP	PT, DE	INESC TEC
	SUC 6	Day-ahead congestion forecasting	DE AP, DE RP	DE	VITO

	SUC 7	Voltage control in LV networks based on limited observability and network topology	DE AP	PT, DE	INESC TEC
	SUC 8	LV flexibility needs assessment for voltage and congestion management	DE AP, DE RP	DE	KUL
	SUC 12	Minimizing costs linked to DSO flexibility requirements	DE AP, DE RP	DE	N-SIDE
Flexibility Market	SUC 13	Short-term flexibility procurement	DE AP, DE RP	PL, PT, DE	NODES
Flexibility Aggregation	SUC 16	DER registration and configuration	DE AP, DE RP	PT, DE	CENTRICA
	SUC 17	Bidding aggregation	DE AP, DE RP	PT, DE	CENTRICA
	SUC18	Resources' dispatch and monitoring	DE AP, DE RP	PT, DE	CENTRICA
	SUC 19	Baselining	DE AP, DE RP	PT, DE	CENTRICA
	SUC 20	Collecting and publishing metering data	DE AP, DE RP	PT, DE	CENTRICA
Data Manage ment	SUC 21	DSO data management – German Demonstrator	DE AP, DE RP	DE	MNS/ E.ON

2.3.3 Key Performance Indicators

The KPIs that will be used to assess the demo results are shown in (Table 2.4) A first set of KPIs concerns the common demo KPIs, indicated by the IDs 'CM_KPI_x'. These KPIs will be evaluated within the German demonstrator, as well as within the other demonstrators of the EUniversal project. Within WP6, a set of demo specific KPIs were defined that are indicated by the IDs 'DE_KPI_x' in Table 2.4. Note that, according to D6.3, an extra demo specific KPI was defined, i.e. DE_KPI_06, with the objective to evaluate the performance of the flexibility needs assessment functionality.

The objective of this deliverable is to describe the set-up of the complete flexibility value chain within the demonstrator and to give the results of the functional tests of each individual tool and market component. These functional tests are necessary to make sure that the KPIs defined in WP6 can be calculated.

However, the complete KPI results based on the evaluation of the full flexibility value chain will be presented in the following and conclusive deliverable on the German demonstrator (D8.3).

Table 2.4 - KPI's relevant to the German Demonstrator

KPI ID	**KPI Name**
CM_KPI_1	Flexible capacity vs. flexible volume offered ratio
CM_KPI_2	Flex volume offered by FSP vs. Flex request by DSO
CM_KPI_3	Flex bids accepted by DSO vs flex volume delivered by FSP
CM_KPI_4	Avoided restrictions
CM_KPI_5	Voltage Magnitude Prediction Error
DE_KPI_01	Costs of Congestion Management with flex Market vs. Curtailment
DE_KPI_02	Cycle Time DSO process
DE_KPI_03	Share of correctly forecasted congestions
DE_KPI_04	Share of false positive congestion forecasts
DE_KPI_05	Baseline accuracy
DE_KPI_06	Over-/under-estimation of flexibility

3 Field test framework

3.1 System architecture

All the system processes and data flows are subdivided into several logical levels: system and data access, data collecting, data processing, internal and external data exchange.

3.1.1 Goals and objectives

The German Demo System was designed and implemented to overcome challenges of data collected from various sources and securely exchanging data while ensuring the independent operation of different applications and tools on the cloud virtual server.

The primary objectives of the German Demo System in terms of functionality are the following:

- setting up flexible a data storage and management, ie. a dedicated storage capacity on the cloud server's local disks + relational database system, to store data from external sources and calculation results from tools.
- building, deploying and managing multiple tools provided by EUniversal Consortium partners for the German Demonstrator
- building, deploying and managing an orchestrating synchronizing application (further – the Synchronizer) with built-in REST API, enabling data exchange between data sources and tools running on the server
- establishing reliable and stable communication between all system components
- maintaining a dockerized development system environment
- centralized logging to track the operation status of the system
- deploying API gateway by means of load balancing and reverse proxying software to support secure processing and handling of HTTP APIs built into the Synchronizer and other applications/tools.

3.1.2 Design principles

The system architecture meets all the primary objectives and needs of the German Demonstrator in the EUniversal Project and provides its assured functionality considering data protection requirements and operational reliability.

The system architecture is based on the following design principles:

- establishing trustworthy communication between all applications and tools within the German Demonstrator
- information protection in data exchange and storage – authentication and authorization provided
- containerized architecture as the basis for system architecture

- separate Docker container for each application/tool/database system
- central orchestrating unit (backend application build on the .NET framework) for data handling, validation and data exchange between all system components
- each application and tool are running independently and have their own specific functionality
- NGINX as general high-performance routing part handling multiple HTTP connections
- predefined scheduled and/or manually triggered data querying from external sources
- synchronous data processing by applications and tools according to a predefined calculation schedule
- capability of database polling through asynchronous queries to ensure better performance
- remote access to logging messages for tool developers to track the tools' status during the testing and operation phase
- Capability of querying tools' calculation results through the Synchronizer REST API along with their visualization and analysis in a BI tool

3.1.3 System Architecture: General Model

The main purpose of system architecture is to determine the essential structural components of the system along with the components' interaction.

The EUniversal system architecture model is shown in Figure 3.1 visualizing the following system component groups:

- external sources supplying data for calculation and analysis (Measurements Platform, Weather Forecast Source, DSO Internal Platform)
- embedded interfaces (CLI) and communication channels to external data sources
- data synchronizer – data processing and orchestration unit
- dockerized applications & tools – data analysis and calculation units
- system users (advanced users and system administrators)
- knowledge database system for data collection

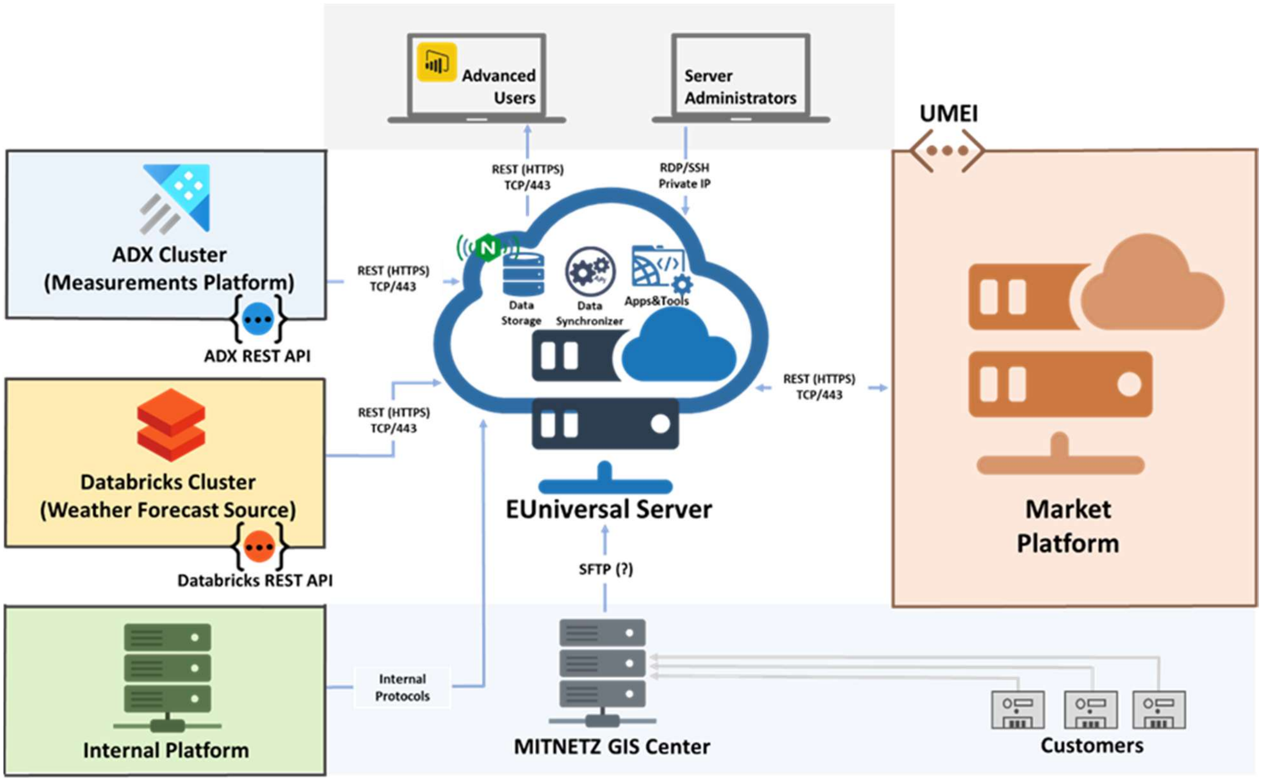


Figure 3.1- System Architecture Model

Each system component group is assigned a specific role (Table 3.1):

Table 3.1- System Components and Roles

System Role	System Component Group	System Component	Role Description
Data Owner		German Demonstrator customers	An organization or individuals that provides data by agreement and give permission for its use with the assurance of confidentiality and pre-anonymization
		DSO	
Data Provider	External data sources incl. communication channels	Azure Data Explorer Cluster	Platform or data source that independently handles data, makes the data available for regular polling through dedicated interfaces, participates in data exchange process on demand of the data orchestrator
		Databricks Cluster	
		Internal Platform	
		NODES Platform	

		Universal Market Enabling Interface	
Data Orchestrator	Data Synchronizer	Synchronizer	Special application in charge of data managing, that is data handling from multiple data sources, as well as data combining, validating, and converting it when necessary, storing the data in a dedicated data storage location (database or a separate local space on the server) and making the data available for Data calculation and tool analysis
Data Calculation and Tool Analysis	Dockerized applications and tools	Congestion Forecasting and Prevention Tool	Continuously performing calculations/processing iteratively a new set of data from external sources; fulfills one of the key functions in calculation chain and produces data that can be entered into the next level of the chain or be handled as a final calculation result
		Optimal Bid Recommender	
		State Estimator	
		Flexibility Needs Assessor	
Data Storage	Knowledge database system	PostgreSQL	A database or dedicated storage space on the server for data collection, storage and providing on demand when receiving a request from a main participant via a data orchestrator
		Local storage	
Data User	System User	System administrator	The person in charge of monitoring the status and performance of the system and making all necessary updates to the system
		Advanced Users	The person in charge of analyzing and visualizing the data coming from the system, and making decisions accordingly

Interaction between system components is ensured by secure communication channels. The communication between data providers and data orchestrator is mainly implemented by polling external APIs of the providers by the Orchestrator on a pre-planned schedule (every 15 minutes or hourly). The data querying from external sources is facilitated through the utilization of standard HTTP and HTTPS protocols, along with the corresponding RESTful API and CLI. The communication between the Orchestrator and Data Collector is supported by message-based protocol. Data exchange between the Orchestrator and corresponding tools/applications running internally on the server performed via querying of the intended internal APIs.

EUniversal UMEI

3.1.4 System Architecture: Dataflow

The need to regularly provide all the tools with sufficient up-to-date input data to perform calculations has led to the integration of an automated data collection system enabling the data acquisition (polling) according to the predetermined time schedule. For an effective management of the system, it's also critical to validate the source data each time it arrives. Considering this, the functions of data query, verification and storage in the system were delegated to the Data Synchronizer.

The dataflow taking place in the system is illustrated in Figure 3.2 and involves the following data and their related sources (Table 3.2):

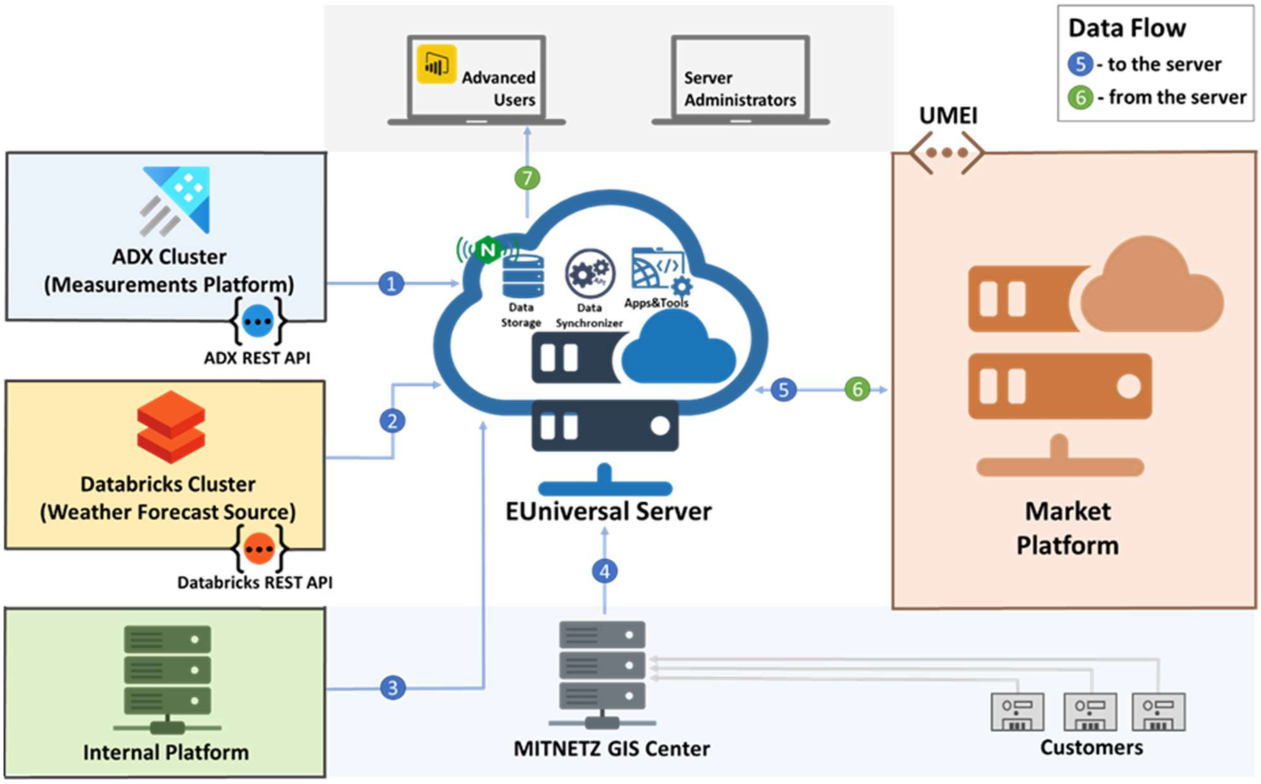


Figure 3.2 - System Dataflow

Table 3.2 - System Data Sources and Dataflow

Nr	Data Recipient	Data Source	Data Source Role	Transmitted Data	Data Description	Data Type	Data Cycle
1	Data Synchronizer	Azure Data Explorer Cluster	Measurement Platform	Node location, line current, voltage, active/ reactive/	Information about all measured values of current,	.json	~ 15 min

				apparent power	voltage and power for each particular node of the network		
2	Data Synchronizer	Azure Databricks Cluster	Weather forecast source	Hourly weather forecasts of temperature and cloud fraction	Predicted values of air temperature and cloud fraction for the location of each node of the network (for each hour for next 28 hours)	.rds	Updated every 3 hour
3		Internal Platform	Network Topology System	Network topology	Information about network structure	.dgs	As updated
4		MITNETZ GIS Center	Measurement Platform	Active and reactive power measurements	Information from smart meters (current, voltage, active and reactive power) installed on customers side	.json	~ 15 min
5		NODES Platform	Flexibility Market Platform	Existing assets, grid assignments, grid areas, grid nodes, energy market, meter reading, price areas, orders, trades	Information about existing assets, grid areas and nodes, market platforms, placed orders, available flexibility resources, orders and trades	.json	hourly
6	NODES Platform	Data Synchronizer	System Operator placing	"Buy" flexibility orders	Information about needed energy volumes (for	.json	As congestion occurs/ predicted

			flexibility orders		each congested node) for mitigating network congestions		
7	System Users	Data Synchronizer	Source of data for analytics and decision making	Calculation results for analytics and visualization	Calculation results per tool, information about calculations' performance, raw measurements fed into the tools	.json	As updated

3.1.5 System Architecture: Internal View and Data Exchange

The internal system view is illustrated in Figure 3.3.

Internal data exchange takes place between the following system components:

- knowledge database system (incl. object-relational database system and dedicated local storage space)
- dockerized calculation tools
- dockerized data synchronizer

The Data Synchronizer is the central orchestrating part of the system that combines all processes of data handling, processing, and transmission to the other system components. Each tool is deployed in a separate Docker container, which allows independent and secure operation of the system. Transferring data from one dockerized component to another is carried out via independent Docker volumes or via the internal IP address of a particular Docker container by means of the Data Synchronizer. Interaction between applications and the knowledge database system is also ensured through the Synchronizer.

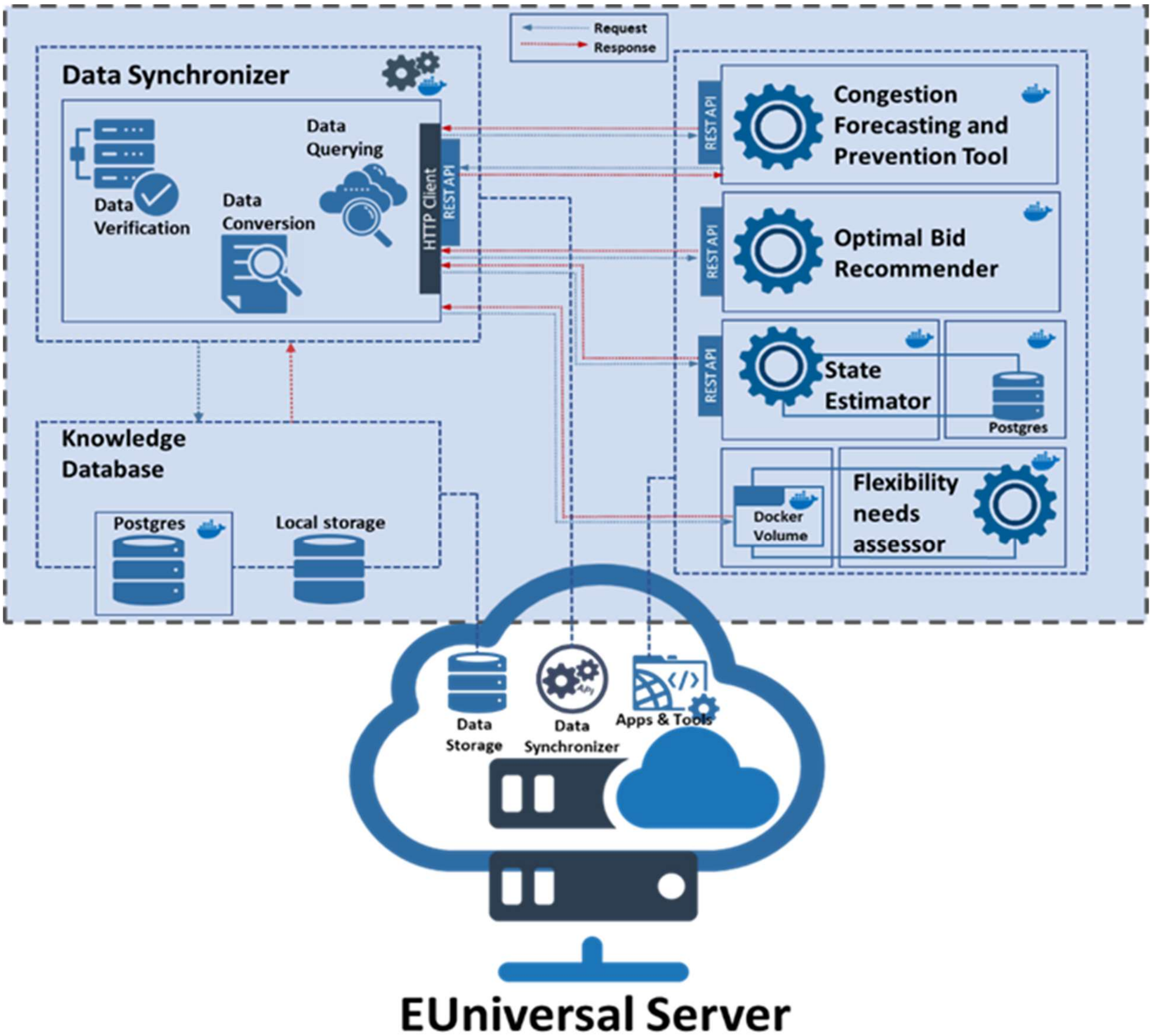


Figure 3.3 - **System Internal View**

Tool Deployment

Given the dockerized development environment as a critical point in the server architecture, the basis for deployment of applications and tools on the server are Docker images and Dockerfiles, which are provided via a dedicated Docker Hub, third-party registry services or pushed directly to the DSO system via a trusted and secure communication channel.

Dockerfile is a premise that contains a set of instructions i.e., statements to assemble a Docker image. In other words, the Dockerfile includes all the necessary build context from which a Docker image is supposed to be built, including all pre-configured environments, code and dependencies.

After creating a Docker image from a Dockerfile or getting one directly from the Docker Hub or Docker repository, the next step is to start a Docker container containing the corresponding application or tool supplied by the image. Once the Docker container is successfully started, it becomes manageable and interoperable with other containers on the server. In this way all applications on the server are deployed and started.

Internal Data Exchange

For establishing a secure and stable interaction with the system, all tools are equipped with a built-in REST API. Some tools are also equipped with an access control mechanism with bearer authentication. Thus, the relevant HTTP requests to a specific API of a particular tool can be used for authentication, entering input data, calculations triggering, monitoring the status of calculations, and finally for the retrieval of calculation results. One of the tools running on the server supports two-way data exchange by automatically pushing calculation results to the Synchronizer via a corresponding POST HTTP request to the REST API of the Synchronizer, which makes the data exchange even more flexible and faster.

Generally, all the main requests for entering input data into a particular tool, triggering and monitoring calculations, as well as extracting results are performed by the Synchronizer through the pre-established HTTP Client instance with the appropriate configuration, headers, and credentials. Thus, the Synchronizer acts as an orchestrator – monitors the data processing and redirects data from one tool to another or from one tool to the knowledge database system. In other words, all the inputs and outcomes are first verified by the Synchronizer (and stored in the knowledge database if needed) before being provided to the next tool in the chain.

3.1.6 System Architecture: Calculation chain

The main purpose of the calculation chain is to evaluate the state of the DSO's network, to identify network bottlenecks, to predict congestions using probability calculation and finally to determine network nodes with available flexibilities to solve the predicted congestions. The flexibility procurement is supported by the market platform and the UMEI enabling data exchange between all participants of the flexibility market.

The Calculation chain is illustrated in Figure 3.4.

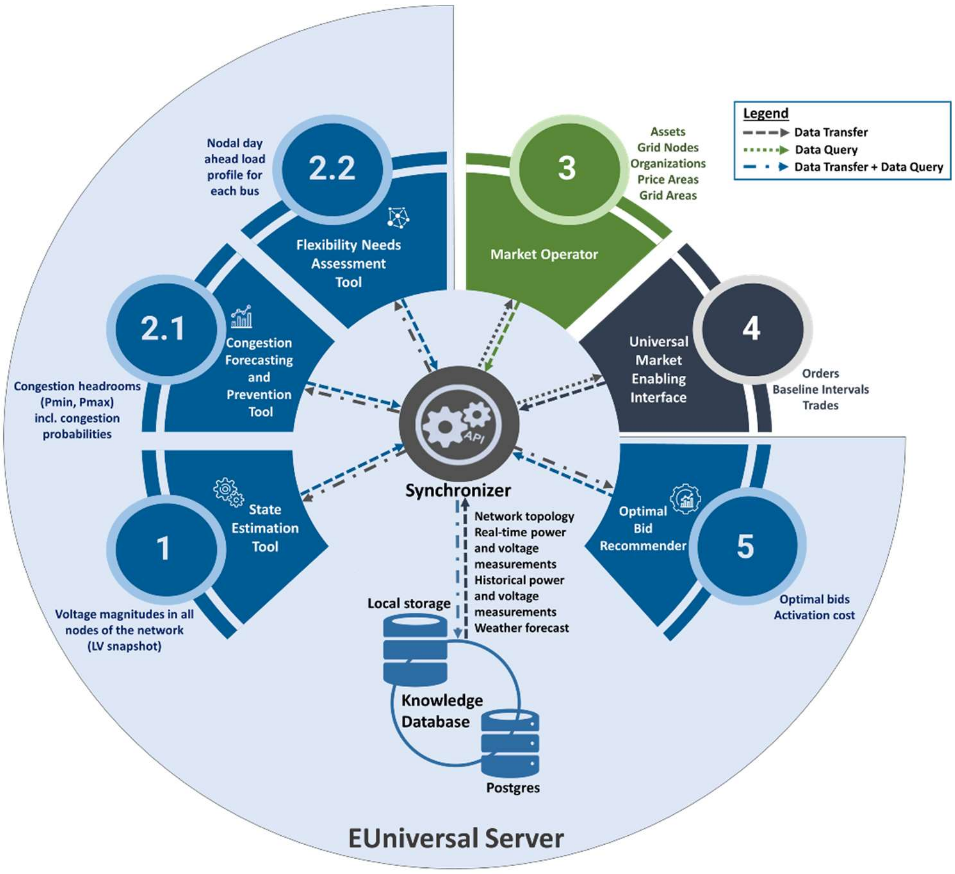


Figure 3.4 **- Calculation chain**

3.2 Internal and external components of the calculation chain

- Congestion Forecasting and Prevention Tool (Ch. 3.2.1)
- Flexibility Needs Assessment Tool (Ch. 3.2.2)
- State Estimator (Ch. 3.2.3)
- Optimal Bid Recommender (Ch. 3.2.4)
- Universal Market Enabling Interface (Ch. 3.2.5)
- Market Operator (Ch. 3.2.6)

The procedure of data collection, processing and calculation includes five basic steps:

- Initial data collection and network state estimation
- Congestion forecasting and flexibilities assessment in the network
- Pre-qualification stage - acquisition of up-to-date market-related information
- Acquisition of up-to-date information about baseline intervals and flexibility orders
- Optimal bids calculation – decision making on orders selection in accordance with offered market flexibilities and predicted congestions

The calculation chain is designed to perform a complete calculation cycle for a set of input data that is entered into the system from every 15 minutes to every hour.

The initial input data required to start the calculations include:

- network topology
- real-time power and voltage measurements at network nodes (for last 15 minutes)
- historical data including power and voltage measurements at network nodes (for last month)
- weather forecasts (day ahead)

3.2.1 Calculation Chain: Congestion Forecasting and Prevention Tool

Location: internal, running on the server.

Tool developer: VITO.

Tool objective: evaluation of network bottlenecks based on input data and prediction of network congestions and probabilities.

Tool triggering: hourly.

Input data

The tool accepts the following input data:

Table 3.3 - Input data Congestion Forecasting and Prevention Tool

Input data	Description	Duration	Sampling
Voltage measurements	Frequency; line and phase voltages.	1 day	Sampled every 15 minutes
Power measurements	Active, reactive, apparent power for each phase; current for each phase.	1 day	Sampled every 15 minutes
Weather forecast	Temperature and cloud fraction for each node.	Last 24 hours	Sampled every 15 minutes
Network structure	Input data during the initialization phase and thereafter in case of the network changes.	-	-

Output data: congestion headrooms with congestion probabilities for next 48 hours.

The tool's operation algorithm is presented in Figure 3.5.

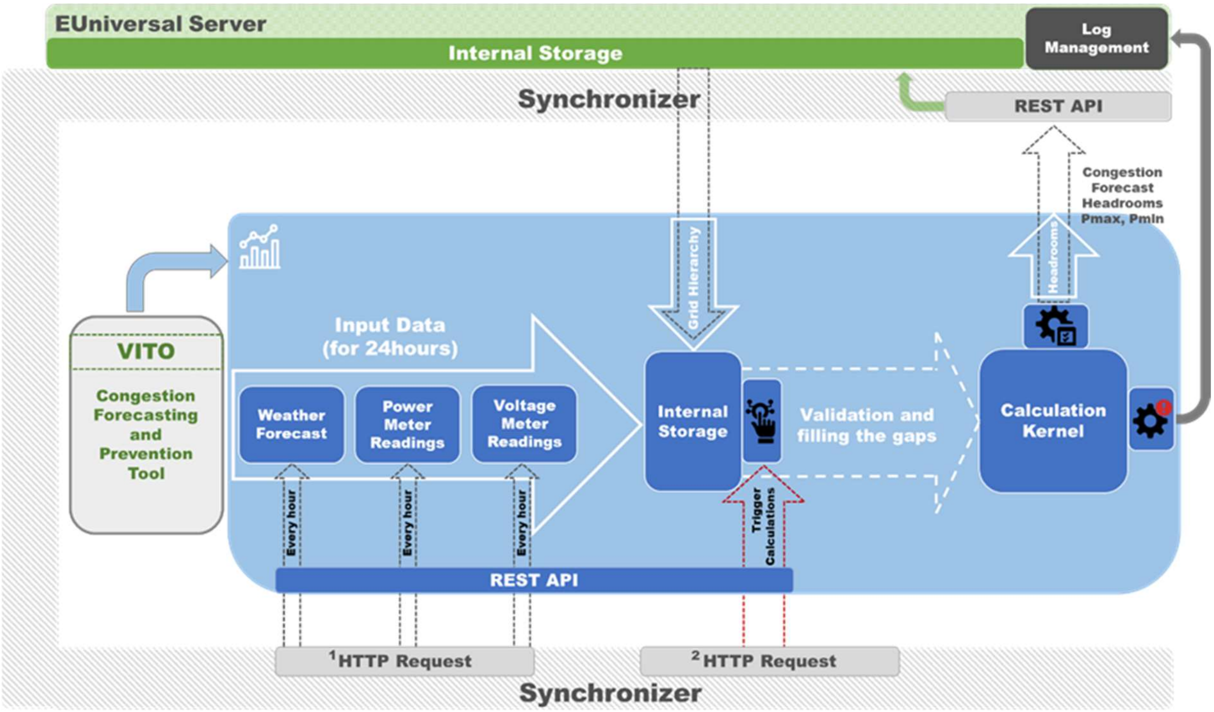


Figure 3.5 - Calculation chain: Congestion Forecasting and Prevention Tool

The input data are entered by the Synchronizer via appropriate HTTP GET request to the tool's REST API. After the data are collected in the tool's internal storage space it is processed by the tool after reception of an additional HTTP GET request to trigger the calculation. The calculation results are automatically pushed back to Synchronizer API via a corresponding HTTP POST request as soon as the calculation is finished. The output data are collected in the intended directory on the server and are available for retrieval through the REST API of the Synchronizer.

Monitoring of events occurring in the tool and tracing of errors and warnings in the calculation process is fulfilled via a proper analysis of log messages generated by the tool and collected in logs repository on the EUniversal server. Tool developers can retrieve the logs of the respective tool via the Synchronizer REST API.

3.2.2 Calculation Chain: Flexibility Needs Assessor

Location: internal, running on the server.

Tool developer: KU Leuven.

Tool objective: providing nodal and zonal FNA required to solve LV network voltage and thermal congestion while reducing distribution network imbalances.

Tool triggering: daily.

Input data

The tool accepts the following input data:

Table 3.4 - Input Data Flexibility Needs Assessor

Input data	Description	Duration	Sampling
Nodal Load Data	Line and phase voltage; line and phase power.	1 month	Sampled every 15 minutes
Grid topology	Grid hierarchical structure (incl. branches, buses and switches) for German Demo networks in DigSilent format	-	-

Output data: nodal day ahead load profile for each bus.

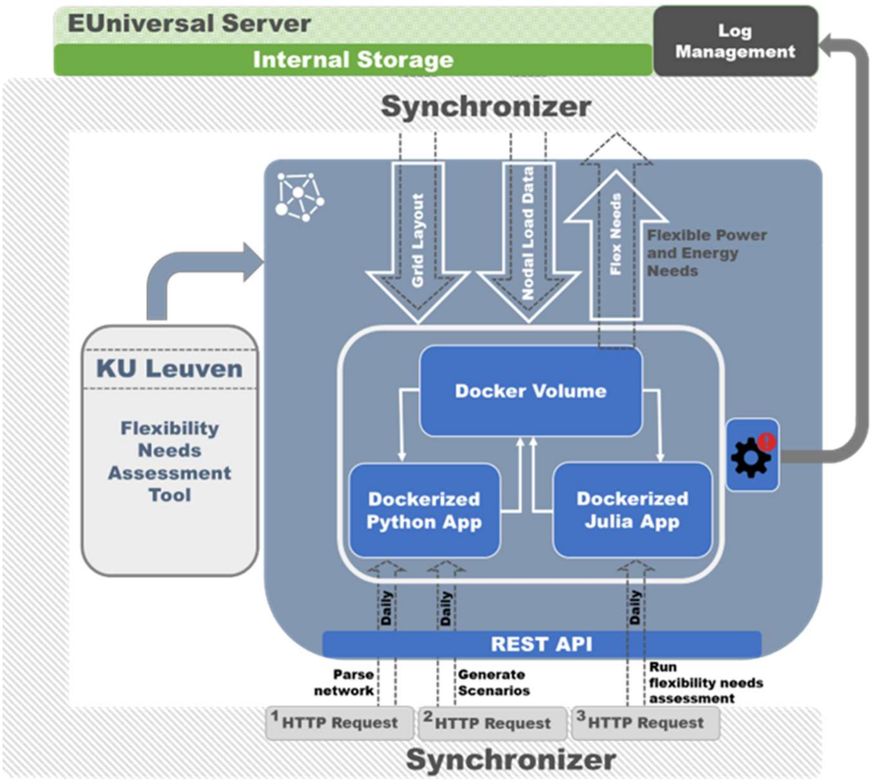


***Figure 3.6* – Calculation chain: Flexibility Needs Assessor**

The FNA Tool is composed of two applications in Python and Julia programming languages. Before running the flexibility needs assessment with the FNA Tool each network provided by the German Demo must be converted from DigSilent format to Json format. To start the network conversion the Synchronizer makes a corresponding HTTP GET request to the FNA API. After receiving a successful response, the Synchronizer makes the next HTTP GET request to generate scenarios for grid assessment based on the available nodal load data. Both HTTP requests are made to the Python application. A successful response to the second request means the successful generation of scenarios, which are stored in a Docker volume shared by both Python and Julia applications. The final FNA execution request is performed by the Synchronizer to the Julia application, which in turn fetches the calculated scenarios from the Docker volume and makes a flexibility needs assessment for the German Demo networks. The assessment outputs are stored back in the Docker volume and then extracted by the Synchronizer.

Monitoring of events occurring in the tool and tracing of errors and warnings in the calculation process is fulfilled via a proper analysis of log messages generated by the tool and collected in logs repository on the EUniversal server. Tool developers can retrieve the logs of the respective tool via the Synchronizer REST API.

3.2.3 Calculation Chain: State Estimator

The State Estimator is the first tool in the calculation chain.

Location: internal, running on the server.

Tool developer: INESC TEC.

Tool objective: Evaluation of the voltage magnitudes for the network nodes where no real-time measurements are available.

Tool triggering: every 30 minutes.

Input data

The tool accepts the following input data:

Table 3.5 - Input Data State Estimator

Input data	Description	Duration	Sampling
Historical measurements	Voltage and active power measurements; required for each node for initialization.	1 month	Sampled every 15 minutes
Real-time measurements	Voltage and active power measurements for nodes	15 minutes	Sampled every 15 minutes

Output data: voltages in all nodes of the network including nodes with no real-time measurements for every 15 minutes.

The tool's operation algorithm is presented in Figure 3.7.

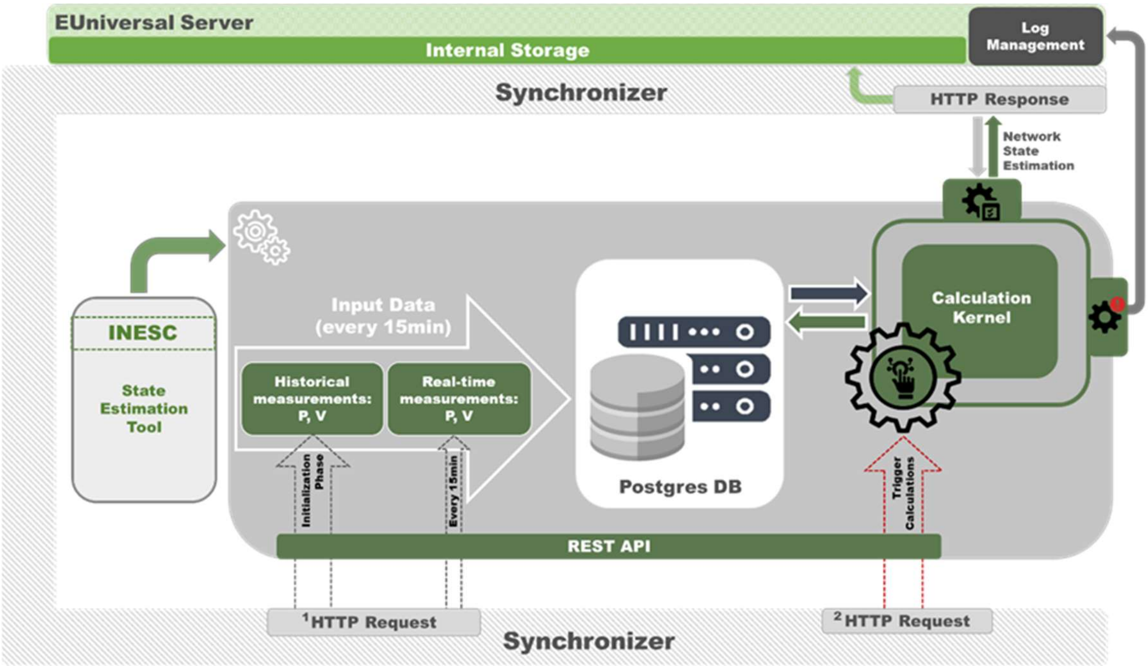


Figure 3.7 - Calculation chain: State Estimator

During the initialization phase, historical data, including voltage and power measurements for each node of the network, are entered into the tool by making a corresponding HTTP GET request to the tool's API. Next, the real-time measurements on each initialized node are sent to the tool every 15 minutes by making another HTTP GET request. All input data are stored in the database (Postgres), which is linked to the tool and runs in a separate Docker container. The calculations are triggered every 15 minutes by an HTTP GET request and the calculation results are returned consequently. The output data are collected in the intended directory on the server and are available for retrieval through the REST API of the Synchronizer.

Monitoring of events occurring in the tool and tracing of errors and warnings in the calculation process is fulfilled via a proper analysis of log messages generated by the tool and collected in logs repository on the EUniversal server. Tool developers can retrieve the logs of the respective tool via the Synchronizer REST API.

3.2.4 Calculation Chain: Optimal Bid Recommender

Location: internal, running on the server.

Tool developer: N-SIDE.

Tool objective: calculation of accepted bids and activation prices in accordance with given congestions and submitted flexibility orders.

Tool triggering: hourly.

Input data

The tool accepts the following input data:

Table 3.6 – Input Data Optimal Bid Recommender

Input data	Description	Duration	Sampling
Congestion Headrooms	Predicted congestion headrooms provided by the Congestion Forecasting and Prevention Tool	Next 48 hours	Sampled every 15 minutes
Flexibility orders	Flexibility orders submitted by the FSP on the market platform	Next 48 hours	Sampled every 15 minutes
Baseline intervals	Actual power consumption for each node of the network submitted by the FSP on the market platform	Next 48 hours	Sampled every 15 minutes
Network structure	Information about network hierarchy with parent and child nodes	Next 48 hours	-

Output data: accepted bids and activation prices related to flexibility orders submitted on the market platform.

The tool's operation algorithm is presented in the Figure 3.8.

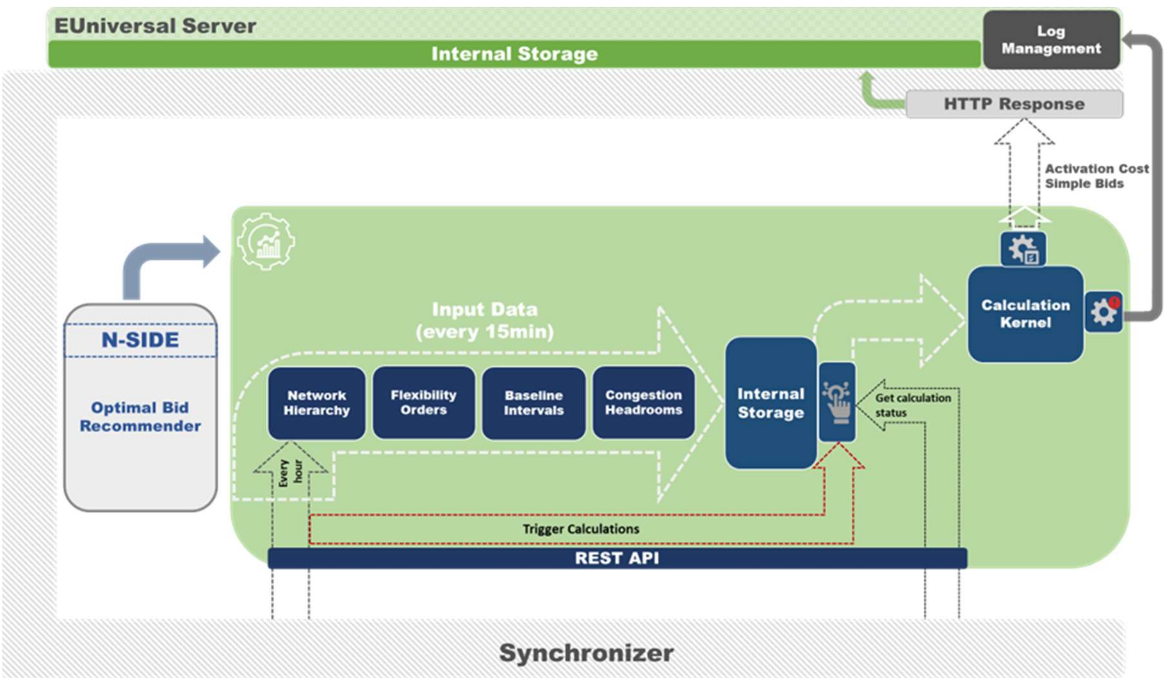


***Figure 3.8* - Calculation chain: Optimal Bid Recommender**

The final step in the calculation chain refers to the Optimal Bid Recommender. After the calculation of the network congestions have been made and the updated information of submitted sell orders and baseline intervals are collected from the market platform, all this data is fed into the tool to produce results on suitable orders and their activation prices according to the predicted congestions. Calculation results serve as the basis for DSO's decisions regarding order submission on the flexibility market to mitigate network congestions. The input data are entered into the tool by the Synchronizer via appropriate HTTP GET request to the tool's REST API, to then initiate the calculation. The calculation status is checked by the separate HTTP GET request. After the calculation status is "completed", the results of calculations are polled by another HTTP GET request, then collected in the intended directory on the server and are available for retrieval through the REST API of the Synchronizer.

Monitoring of events occurring in the tool and tracing of errors and warnings in the calculation process is fulfilled via a proper analysis of log messages generated by the tool and collected in logs repository on the EUniversal server. The tool developers can retrieve the logs of the respective tool via the Synchronizer REST API.

3.2.5 Calculation Chain: Market Platform

Market Operator: NODES

Location: external.

Objective: providing the DSO with all the necessary information and means to flexibility orders on the market platform.

Requesting: daily or after making changes in the DSO's network structure.

Requested data:

- Asset Types;
- Grid Nodes;
- Grid Areas;
- Organizations;
- Price Areas;
- Grid Hierarchy;
- Time Zones.

The algorithm for data requests by the Synchronizer via the NODES API invocation is shown in Figure 3.9.

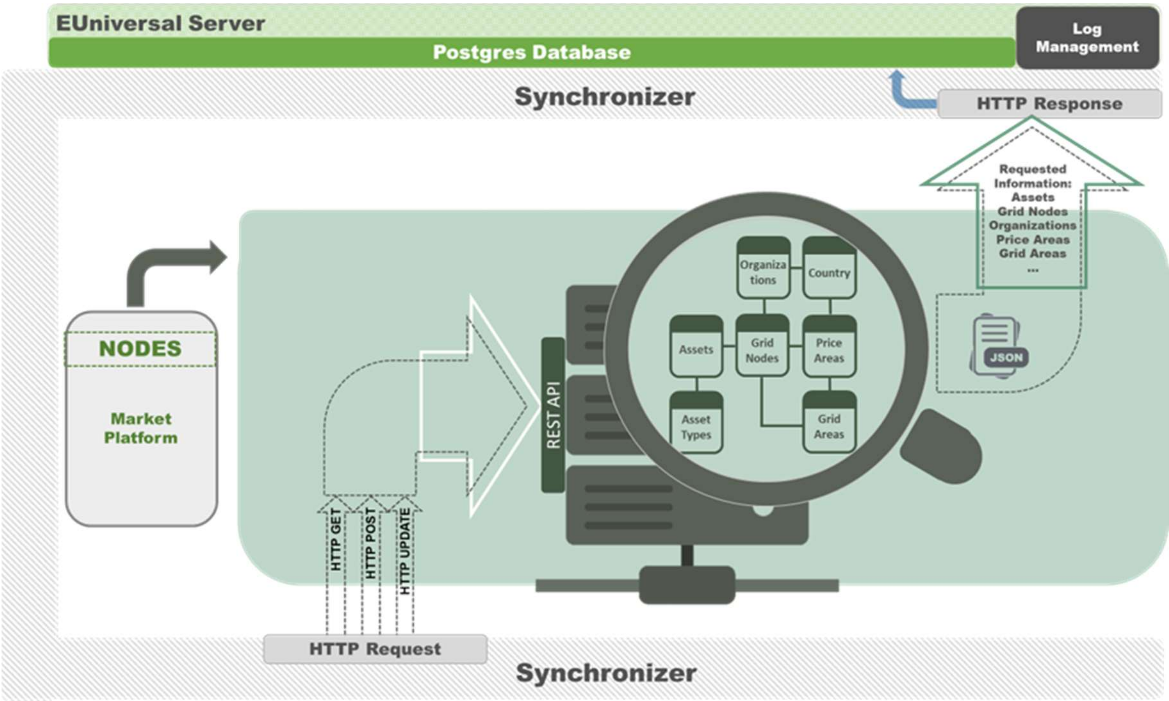


***Figure 3.9* - Calculation chain: Market Platform**

The NODES Market is queried daily by the Synchronizer through the UMEI to check the market platform for recent updates. Besides, the queries are performed after changes have been made to the network structure (adding a new node, deleting an invalid node, updating the information about a particular node, etc.). For these purposes, NODES API provides all necessary conventions and request types that allow changes to the DSO's network structure directly through the API. The data provided in the responses are stored in the intended database on the server and then used for order submission via the UMEI API.

3.2.6 Calculation Chain: UMEI API

Developer: NODES, N-SIDE, Centrica.

Location: external.

Objective: establishing communication between the DSO and other market participants by exchanging up-to-date information about submitted baseline intervals, orders, completed bids and enabling the placement of flexibility orders by the DSO.

Requesting: hourly.

Requested Data (for an appropriate time period):

- Baseline Intervals;
- Submitted flexibility (buy/sell) orders;
- Current status of relevant trades.

The algorithm for data requesting by the Synchronizer via the UMEI API is shown in Figure 3.10.

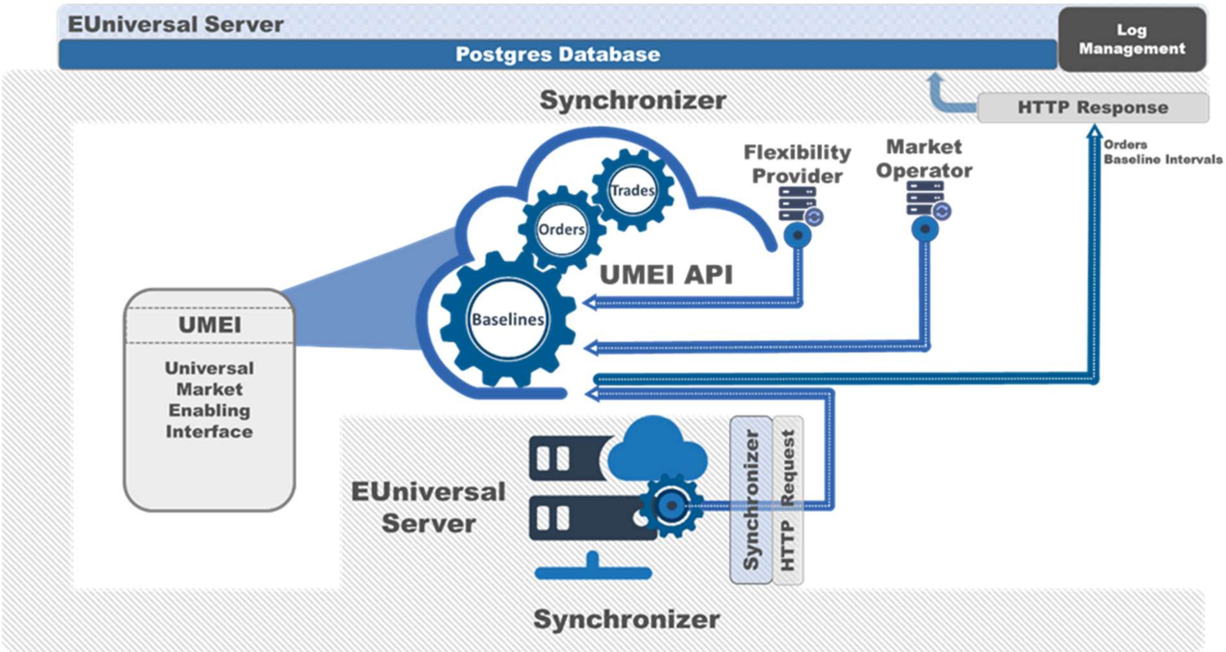


Figure 3.10 **- Calculation chain: UMEI API**

In the fourth step related to the acquisition of appropriate information by the Synchronizer for further calculations the UMEI API acts as an external source. The UMEI API is used hourly to obtain valid orders and trades from the market platform as well as to retrieve baseline intervals submitted by the Flexibility Service Provider (further – the FSP). This information is a necessary input for the Optimal Bid Recommender next in the calculation chain. The appropriate HTTP requests to the UMEI API are performed by the Synchronizer and the corresponding HTTP responses are verified and stored in the Postgres database on the server.

3.3 Customer engagement

3.3.1 Customer acquisition

As neither the smart meter rollout in Germany is far advanced, nor is there any active regulation in Germany on the use of flexibilities in the low voltage grid, which the field tests could follow, it was of great importance to recruit volunteers for the testing. The strategy for customer acquisition was also briefly described in Deliverable 8.1.

The following section further elaborates on the customer topic and discusses the difficulties in the organisation, the mitigation measures and consequences for the field tests and gives an outlook for conclusions that will be further elaborated in the last deliverable of the German demo D8.3.

Barriers

Several challenges were faced during the customer acquisition process, which are briefly described with their effects in the following.

- *Lack of interest in energy/technology topics*

  The current geopolitical situation has made people aware of the importance of a secure energy supply. However, very few contact persons are interested in the technical details. Most of the contacted people did not respond to the enquiries - despite personal letters and reports in the local press.

- *Corona pandemic and constraints*

  Due to the pandemic situation, some citizens stayed away from the town hall meetings. Digital formats were offered as an alternative. However, the personal connection and the possibility to ask questions individually were limited.

- *Lack of financial incentives*

  The participants did not incur any costs and were provided with energy management systems for the test period as an incentive. However, without a recognisable clear and long-term financial incentive, only a few people were willing to sacrifice time to participate in the project and allow their premises to be externally controlled.

- *Already existing energy management set-up*

  Technical savvy citizens partially had their own HEMS set-up already in place with motivation to increase the degree of self-sufficiency with their own PV and storage or to connect smart home applications. This reduced the incentive provided by the HEMS. Besides, problems were feared when restoring the own set-up at the end of the project period.

- *Devices without web-communication interfaces*

  Older inverter types could not be used because they did not yet have web interfaces. As this was largely the case with devices built before 2015, this barrier is expected to disappear in the medium term.

- *Focus on autarky/no external influence desired*

  Some citizens reported a strong interest in being as independent as possible from the public energy supply. The idea of being externally controlled by the DSO or FSPs is clearly opposed to this.

- *Large number of devices, manufacturers and interfaces*

  Since the DSO does not have the data of the device types, the compatibility had to be checked beforehand. Furthermore, as there are no standard interfaces for communication with HEMS so far, most of the devices of interested persons had to be excluded. In addition, there were difficulties in finding service providers in the field, as craft enterprises usually specialise in a few manufacturers of inverters. This meant that an individual set-up was necessary for each customer. In some cases, extra communication modules were missing in the inverters. Sometimes installation codes of the devices were necessary to change settings allowing for external connections.

- *Poor reputation of the energy sector/energy transition*

  A multitude of changes combined with the energy transition make it difficult for customers to evaluate what options are beneficial and what are the consequences. Constantly rising energy prices and grid fees cause resentment.

- *Data protection and IT requirements*

  High IT security and data protection requirements had to be fulfilled for the installation of the energy management systems. The classification and provision of all verifications delayed the implementation process and the provision of a visualisation application for the end customers.

- *Lack of experience in customer attraction*

  Apart from metering services and connection requests, DSOs have limited contact with end customers and usually do not advertise for private customers. As a result, they lack experience in appropriate marketing.

3.3.2 Mitigation Measures

To attract customers to the project, proactive communication, takeover costs and thus free participation of end users, were performed in the acquisition process. Additionally marketing material like a promoting video and townhall events were used to gather interest in the project. When the lack of interest among residents became apparent, further measures were planned.

Additional effort was made with an on-site campaign, which included door advertising in the selected grid regions, as well as promotional leaflets to convince more residents with flexibility resources to participate in the EUniversal project. However, this action did not lead to further participation interest which means that further mitigation measures had to be introduced. These include that potentials of flexibility markets are evaluated based on the number of customers present in the grid regions instead of using the actual volunteering residents. In addition, an internal acquisition of MITNETZ colleagues with inverters and batteries was sought to have more customers for the aggregation in the FSP tests. Table shows the final figures of the participating volunteers.

To still be able to build a portfolio out of the few participants, the local affiliation to the selected test grids was removed and thus the aggregation was separated from the market analysis allowing for conclusions for portfolio set-up and aggregation instead of market liquidity, which would not be representative without regulatory financial incentives. In other words, grid nodes of the EUniversal market analysis will include simulated customers. Instead, the possible market potentials and their effect on congestions in the low-voltage and upstream medium-voltage grids will be evaluated using the developed smart grid tools and measurement as well of meta data of the selected LV grid. Real interaction of FSP with end-users will also be part of the field test but will be evaluated in a separated test case. Results of this analysis will be shown in the final Deliverable 8.3 of the work package.

Table 3.7 - Overview of market liquidity and customers recruited

	MFn4420	MLq0094	MIi809	MITNETZ colleagues
Inverters				
...of which customers with interest	13	7	7	18
...of which compatible (monitoring)	5	1	2	9
...compatible incl. steering	2	0	0	5
Batteries-				
...of which customers with interest	3	1	1	17
...compatible incl. steering	1	0	0	5
Heat Pumps				
...of which customers with interest	2	1	1	5
...compatible incl. steering	0	0	0	0
Heat Storages				
...of which customers with interest	-	2	6	-
...compatible incl. steering	-	-	-	-
Wallboxes (+ EV)				
...of which customers with interest	-	-	-	4
...compatible incl. steering	-	-	-	1

4 Field test protocol

This report outlines the different tests and Demonstrator scenarios performed in task 8.2 to prove the reliability of each tool and tool results, and the operational processes required to comply with each tool's minimal viable product (MVP). Due to the complexity of the individual tools and systems as well as the entire flexibility value chain, in a first step every component of the flexibility value chain was tested individually to ensure a correct functioning and information exchange between connected tools, as well as the required data input and output according to the presented SUCs (Ch. 2.3.2).

4.1 Congestion Detection and Flexibility Need Quantification

In this chapter, the smart grid tools are briefly introduced. For further information, please refer to Deliverables 4.1 and 4.2 for the Smart Grid tools, as well as Deliverable 5.1 regarding market processes.

4.1.1 The Pythia tool: Day-ahead LV Congestion Forecast and Asset Headroom calculation - VITO

As already mentioned in the previous chapter, the first step in the flexibility value chain process is the forecast of possible congestions (day ahead). Forecasting congestions in the Low Voltage (LV) distribution grid is a challenging task, mainly due to the very stochastic behavior of end-consumers, and because the grid layout is partially unknown (namely, the exact phase connectivity of the single-phase connections is unknown).

These challenges are tackled by the LV congestion forecast tool in Pythia by taking a statistical approach. Pythia calculates the probability density of voltage and current levels within the network, using available knowledge on the network, and by considering the stochasticity of unknown variables. The tool assumes that the following aspects of the LV system are known and can be used as input to the tool:

- The LV grid layout
- High-level metadata on end-consumers (yearly offtake, connection capacity, etc.)
- Historic and recent connection profile measurements of a group of representative users (not necessarily connected to the same grid system)
- Weather forecasts
- Information on the flexible assets, as available to the DSO: the type, location, and power rating of the asset.

In a second step, Pythia calculates the available headroom capacity for a secure flexibility activation on the network. This headroom capacity indicates the maximal flexibility that can be activated from the flexible devices without creating congestions for every time step. The headroom capacity drops when the congestion probability increases.

Details on the calculations performed by Pythia can be found in Deliverable D8.1 of the project.

4.1.1.1 Functional tests for input data correction

In a first phase of the German Demonstrator field test, functional tests were performed to make sure that correct and accurate forecast results are produced by the LV congestion forecast.

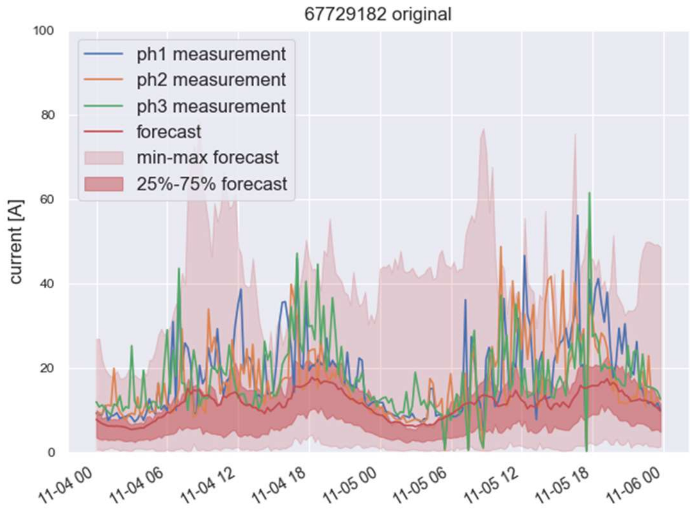
It is obvious that inaccuracies of the input data lead to inaccuracies in the output forecast. The most common inaccuracies found in LV network environments are errors in the grid layout. LV networks mostly consist of (not visible) underground network cables. This easily results in errors while digitalizing network layouts. Secondly, the electricity network is not static. Over time, the LV network changes, e.g., new feeders are added to the system, feeders are switched differently, or new customers are connected, and these changes must also be registered in the digital version of the network. Having an up-to-date digitalized grid layout available is crucial to perform an accurate congestion forecast.

A second source of inaccuracies of the input data is in the representativeness of historic connection profile measurements. If certain user groups are not or under-represented in the set of historical profiles, their electricity offtake pattern will not be captured by the congestion forecast tool and will thus lead to inaccurate congestion forecasts.

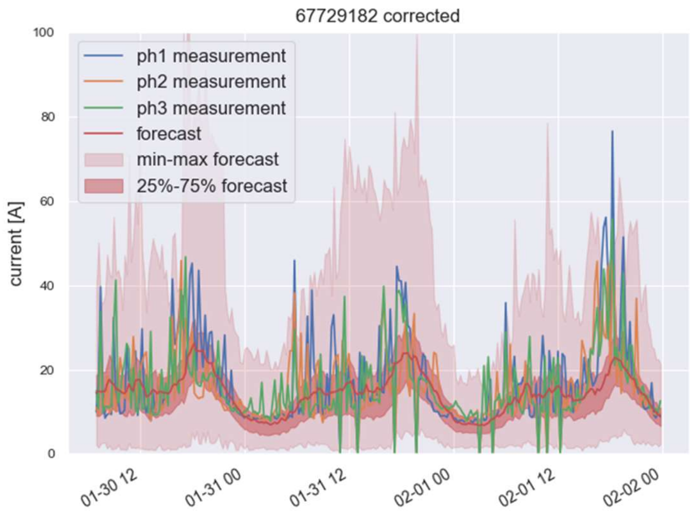
Related to this is the availability of accurate information on end-consumers' characteristics, and their flexible assets. When the offtake pattern of end-consumers changes, e.g., because photovoltaic panels are installed at their premises, their heating system is changed to a heat-pump, or their connection is used to charge an electric vehicle, this information must be known to obtain accurate congestion forecast results.

To evaluate the correct performance of the congestion forecasting tool, the forecasting results were compared with measurements installed in the field.

Figure 4.1 shows such comparison: the currents measured at the head of a specific feeder within the Demonstrator are compared with their forecast. The Pythia LV congestion forecasting tool gives a statistical forecast as result, and therefore the forecast is shown as a range, with a probability density. The congestion forecast using the original input dataset is shown on top (Figure 4.1a). This figure shows that the forecast is structurally underestimating the currents in the network, pointing to missing connections in the network data. After a careful check of the data, it was found that a few household loads were missing in the original data. Correcting for this error resulted in the congestion forecast result shown at the bottom (Figure 4.1b.), where the structural underestimation has disappeared.



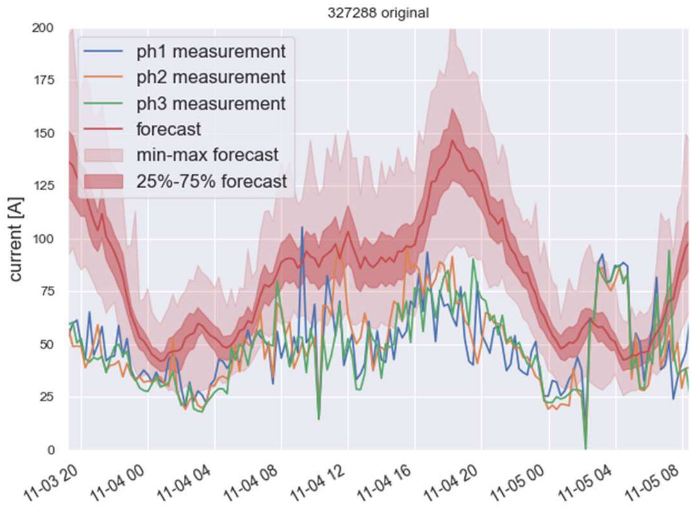
a.



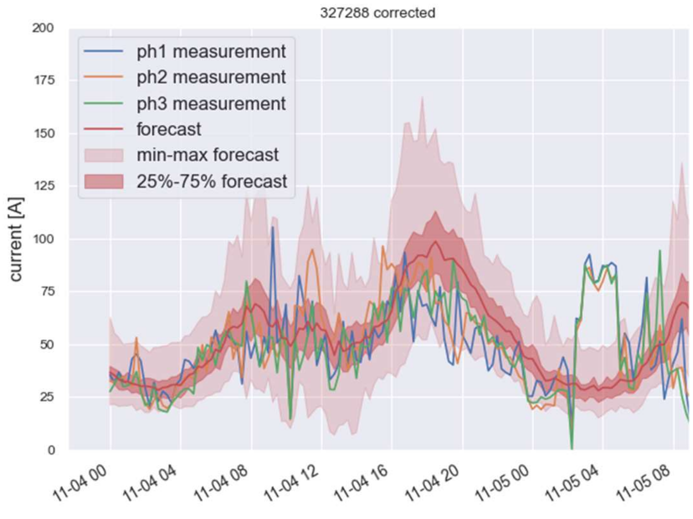
b.

Figure 4.1-Evaluation of congestion forecasting tool: (a) using the original network layout and customer information; (b) using the corrected grid layout and network information for feeder Id. 67729182.

A second example of a case where inaccurate input data lead to inaccurate forecasting results is given in Figure 4.2. Similarly, the figure shows the currents measured at the feeder head, compared with the forecasting result, on top (Figure 4.2a.) using the original data input, and at the bottom (Figure 4.2b.) for the corrected data input. For this case, the original result showed an overestimation of the forecast. The reason for this overestimation was that the historical profiles were not representative enough. More specifically, the feeder supplies a larger commercial consumer, which appeared to be overestimated using the original grouping of historical profiles and customers. Re-grouping of the historical profiles, and network customers according to their characteristics led to better and more representative forecasting result, shown in Figure 4.2b.



a.



b.

Figure 4.2-Evaluation of congestion forecasting tool: (a) using the original network layout and customer information; (b) using the corrected grid layout and network information for feeder Id 327288.

An observation that can be made, when examining Figure 4.2 is the fact that the consumption of a (large) resource is missed in the congestion forecast. This offtake originates from a large electric heating system present in the network. This device is considered a flexible resource in the German Demonstrator and is not included in the forecast of the default consumption presented in Figure 4.2. This first phase of functional testing, where measurements and forecasts were compared, allowed to correct inaccuracies in the overall input data set. This effort also shows that Pythia can also be used to discover such errors in the grid data.

The impact of the flexible devices on the forecast accuracy is discussed in the next section.

4.1.1.2 Including flexibility resources in the LV congestion forecast and headroom calculation

Simultaneous activation of multiple flexible resources on the LV network, e.g., to solve issues on the MV level, could cause congestions on that LV network. Therefore, the available headroom capacity for secure flexibility activation on the network is calculated as a second step in the Pythia flexibility value chain. This headroom capacity indicates the maximal flexibility that can be activated by all flexible devices combined without creating congestions for every time step. The available headroom capacity is reduced when the congestion probability is too high.

The headroom calculation is executed for a worst-case scenario, and thus with the assumption that all flexibility resources are activated at the same time. To be able to do this calculation, the input data must contain the location of the flexible resources and their maximum power. Also, it is important to know whether the resources have a three-phase or single-phase connection, since the network impact of a device with a three-phase connection will be much lower than the impact of a single-phase connected device with the same power level.
The flexibility resources included in the calculation are:

- Heat pumps
- Electric heating systems
- Batteries
- Electric vehicles
- Photovoltaic installations

In Figure 4.3 this process is illustrated: the congestion forecast of the default consumption is shown together with the worst-case flexibility forecast for one feeder of the Demonstrator. In this figure it can be observed that the offtake of the flexibility resource(s) is not captured by the default congestion forecast but is contained within the worst-case flexibility forecast.

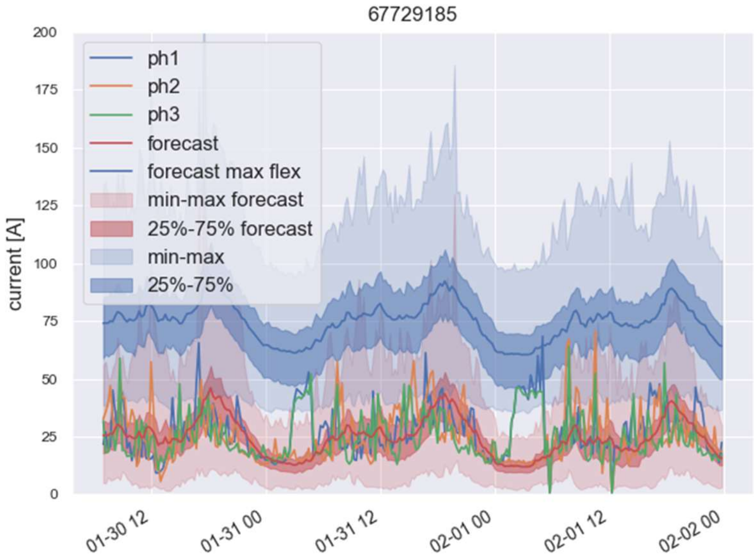


Figure 4.3 - Congestion forecasting of feeder 67729185

- Default consumption forecast, shown in red, and worst-case flexibility forecast, shown in blue.

In a second phase, the resulting worst-case flexibility forecast is transformed into a congestion risk, on which the headroom calculation is based.

The congestion risks are defined as overvoltage, undervoltage or overcurrent congestion risks, defined by the risk that the voltage or current anywhere in the network surpasses a predefined voltage or current limit. The predefined voltage and current limits are set at +/-5% of the rated voltage and at 80% of the current rating, respectively.

Given the existing voltage and current limits, very low congestion risks are found, even for worst-case flexibility situations for every feeder present in the Demonstrator. An illustration is shown in Figure 4.4: the congestion risks for the feeder and measurement period are virtually non-existent, leading to a calculated headroom which always equals the maximum flexibility available in the feeder. As mentioned above, photovoltaic installations are flexible assets since they can be curtailed. How much capacity is available to be curtailed depends on the solar irradiation which is variable throughout the day. Therefore, the maximal available flexibility (and consequently the headroom) can vary over time, also shown in Figure 4.4.

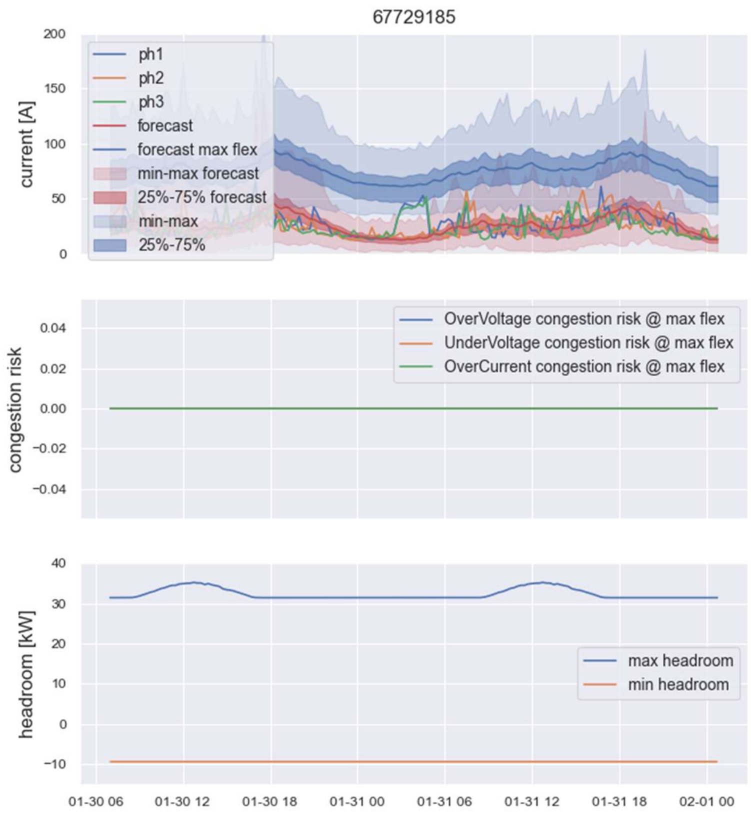


Figure 4.4 - Congestion risks and headroom calculation results for feeder 67729185

In a following phase of the Demonstrator, artificial limits for network voltage and current will be defined such that congestion risks are detected in the Demonstrator-feeders (although in reality the grid is far from congested). These artificial limits will be defined to enable the operation of the full flexibility value chain during the pilot phase.

The choice of the artificial limits, used during the pilot phase, will be elaborated in the following deliverable D8.3, next to a discussion on the pilot measurements and the analysis of the pilot KPIs.

4.1.2 Day-ahead flexibility needs assessment tool development – KUL

The flexibility needs assessment tool tailor-made for the German demo under the framework of the EUniversal project can be divided into the following steps:

- Parsing the grid data [1], [2],
- Creation scenarios using representative load profiles for all the nodes [3],
- Formulation of the optimization problem for quantifying temporal and nodal flexibility needs for avoiding voltage and thermal violations [4], [5], [6], [3], [7], [1],
- Clustering tool for formation of zones [2], and
- Docker implementation of the tool.

4.1.2.1 DigSilent (DGS) Parser and scenario generation

A parser has been created to convert the DGS (DigSilent) file format into a JSON file that is readable to the PowerModels script. The parser is derived from the GridCal python package. The main difference between the two formats is that the DGS data format contains different classes with different information in a hierarchical structure, whereas the JSON file just requires information on the buses, branches, and devices in the grid. For additional details on the parser, refer to [2].

Future uncertainties are considered in the form of scenarios. The scenario generation uses historical data and point forecasts with known forecast errors. In the present case, large amount of historical data is unavailable, instead, a point forecast along with its associated forecast error is known. In such a case, a multivariate Gaussian distribution can be formed by using a point time-series forecast as the mean value and the variance of the distribution is proportional to forecast error [3].

4.1.2.2 Flexibility needs assessment (FNA)

Flexibility needs assessment (FNA)[3] refers to the amount of flexibility the DSO needs to plan or procure from the flexibility market to avoid probable Distribution Network Incidents (DNI). The probable DNI are captured using modeling uncertainties, and using generated scenarios that emulate the different Monte Carlo realizations[4] which could happen. The scenario generation utilizes the nodal load and generation forecast along with historical forecast errors. A flexibility needs assessment-optimal power flow (FNA-OPF) problem is solved for each of the scenarios. The robust FNA, considering the worst-case scenario, if used for flexibility procurement would lead to substantial over-procurement. In order to avoid this, a risk-based index, e.g. a chance constraint (CC), is introduced. Higher values of the CC would project on to greater risk the DSO might have to encounter by facing unresolved DNIs. DNIs in low voltage grids are often local problems in which flexible resources in the proximity may respond to avoid these incidents [4], [3], [7].

[3] The FNA algorithm does not assume the location of flexible resources. Although we assume that flexibility is derived at nodes where there is a load or generation source connected. For all the nodes with no load and generation connected, flexibility (up and down regulation) is assumed to be zero. Flexibility for upregulation (derived from load curtailment or consumption increase) is available only at nodes with loads (positive consumption), while ramp-up flexibility is available not only at nodes with DG (negative consumption) but also nodes where nodal load can be increased by activating flexible loads.

[4] One scenario is one snapshot that could happen. To consider uncertainty, we use large number of scenarios, in the order of 100s. These scenarios are analogous to Monte Carlo realizations of the uncertain parameters.

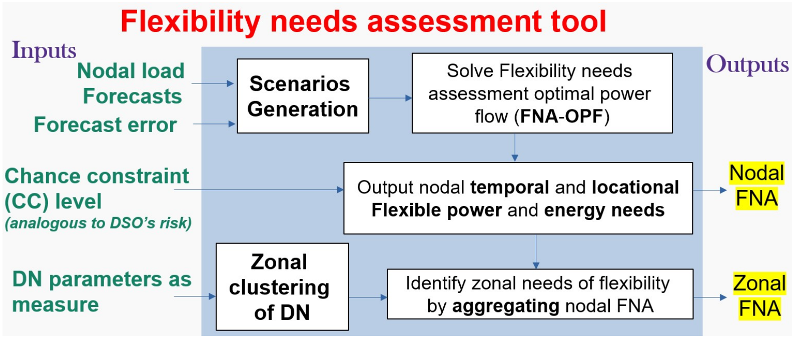


Figure 4.5 – Flexibility needs assessment with the input and the output data

4.1.2.3 Clustering and Zonal FNA

Identifying the zones of an LV DN will help the DSO in planning the flexibility needs of a network. The proposed zone formation solves the following challenges:

a. the formation of connected zones requires an incidence matrix-based measure; therefore, we consider admittance as a measure
b. the admittance matrix cannot be used directly; therefore, spectral decomposition of a doubly stochastic matrix is used, and
c. the selection of the appropriate number of zones is not known a priori.

To apply unsupervised clustering techniques such as k-means, one should know how many clusters are needed. We use the silhouette score as a measure for identifying the best number of clusters [3], [2].

4.1.2.4 Docker implementation

There are three elements that make up the code base. There is code written in Python, in Julia, and everything is implemented to work in one Docker environment. A brief overview will only be provided here and can be seen in Figure 4.6. The primary functions of the python code are:

1. A parser from digsilent files to matpower/powermodels files

2. A scenario generation script

3. Volume for data exchange between docker container #1 and #2

The primary function of the Julia script is to take the grid information and the different scenarios and generate a Nodal day ahead load profile for each bus.

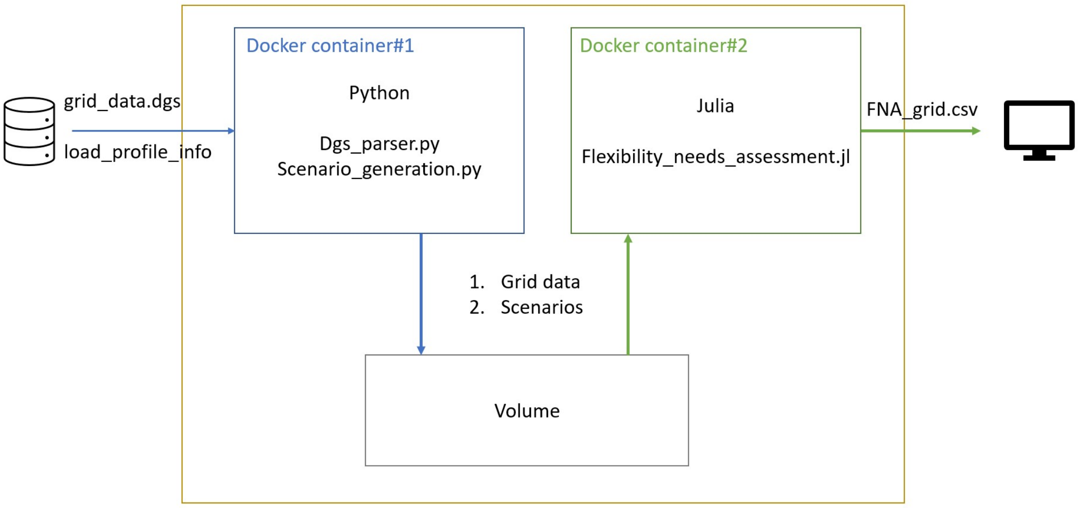


Figure 4.6 - Docker architecture

4.1.2.5 Numerical results for MLq0096

The goal of the case study is to apply the FNA framework for MLq0096. This numerical case study considers a real suburban German DN used for EUniversal demo. This demo network consists of 646 nodes and 331 loads are connected to the DN. Figure 4.7 shows the DN with 7 clusters identified using the DN clustering framework developed under EUniversal project [2].

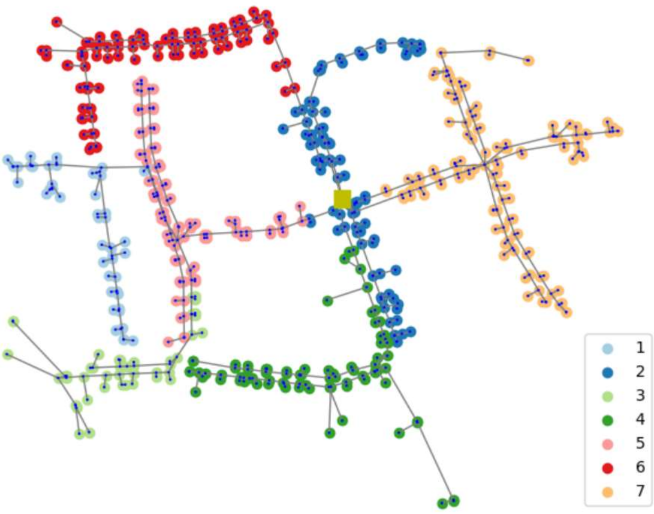


Figure 4.7 - German DN Mlq0094 with zonal clusters

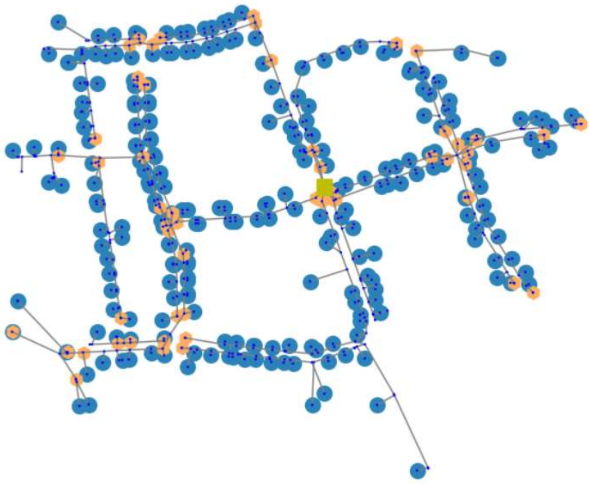


Figure 4.8 - Measurement points (in yellow) and consumers (in blue dots) for MLq0094

The load profiles are selected from a pool of historical load profiles based on prosumer meta-data such as annual kWh, and PV size installed. Based on historical data, the load profile forecast error is assumed to be 30% and PV generation forecast error of 40% is used for scenario generation. Based on these scenarios, the FNA is calculated. The temporal flexibility needs are shown in Figure 4.9. Note from Figure 4.9 that the ramp-up flexibility needs of this DN are both zero, as the PV penetration, in this case, was very low. The total installed PV is 61 kWp and the annual kWh served exceeds 923 MWh.

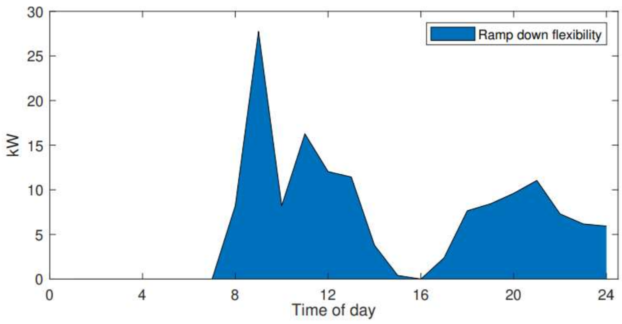


Figure 4.9 - Day-ahead temporal flexibility needs for German DN.

Figure 4.10 shows the impact of chance constraint (CC) on the amount of flexibility needed. The CC level denotes the risk system operator wills to take in flexibility planning. A low value of CC denotes low risk but that leads to a high amount of flexibility needs. The pareto optimal point for these two conflicting goals is shown in Figure 4.10 [3].

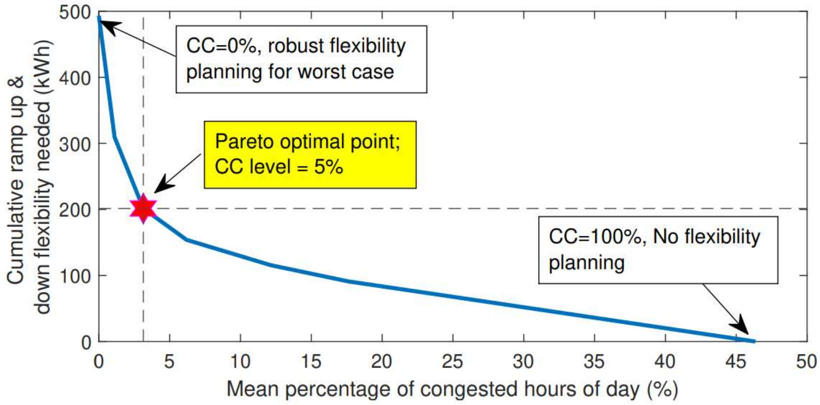


Figure 4.10 - Tuning CC level using Pareto optimality

Figure 4.11 shows the zonal FNA for the DN. Observe that most flexibilities needed are in zone 1, 3, and 4; all these zones are congested at the end of feeders. More than 80% (30.04, 30.45, 19.33% respectively) of the flexibility needed are in these three zones.

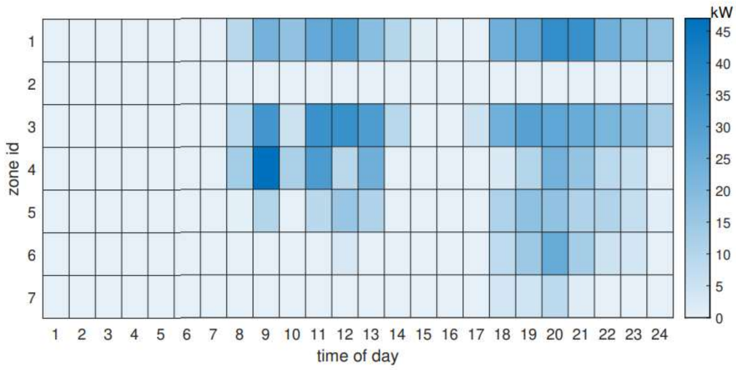


Figure 4.11 - Temporal flexibility needs are calculated in a DA horizon

4.1.2.6 Numerical results for MFn4420

The second demonstration network used is called MFn4420. The zonal clustering, location of measurement points and zonal FNA calculated for MFn4420 are shown in Figure 4.11 - Temporal flexibility needs are calculated in a DA horizon (Figure 4.14).

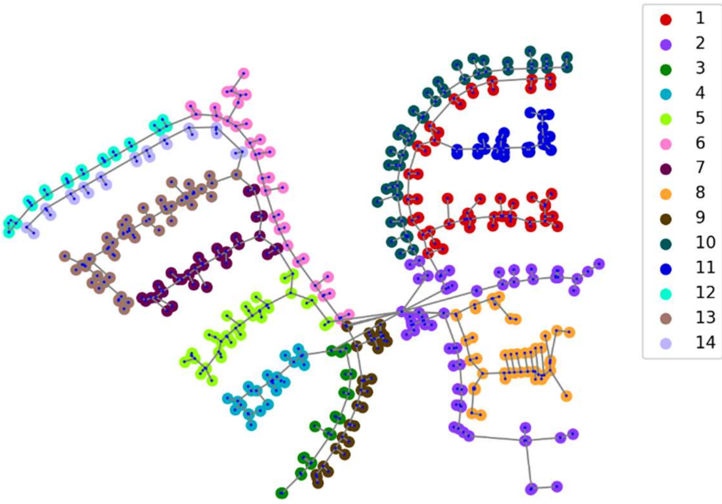


Figure 4.12 - Zonal cluster for demonstration network MFn4420

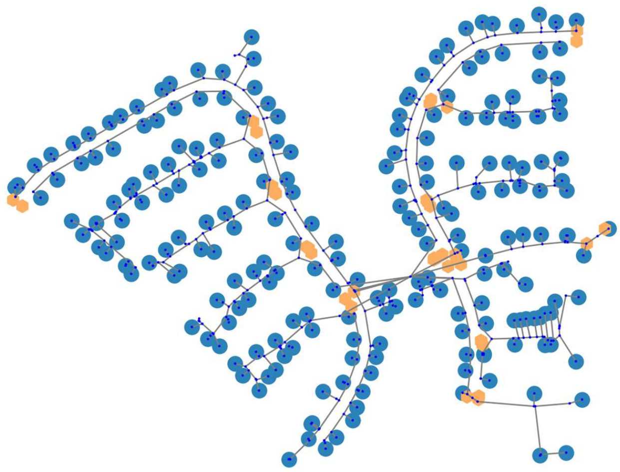


Figure 4.13 - Measurement points (yellow dots) and consumers (blue dots) for MFn4420

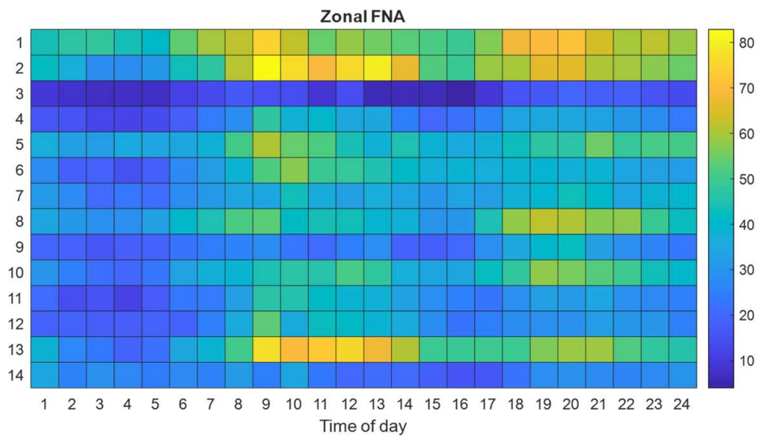


Figure 4.14 - Zonal FNA for MFn4420

Note from Figure 4.14 that most of the flexibility is needed in zones 1, 2, and 13. Since the consumers do not have large amounts of distributed generation in this test case, we observe the flexibility needs in down regulation or load curtailment only. No instances of generation curtailment needs were observed in the simulations.

For both sets of numerical results, the DN congestion is induced by inflating the representative load profiles, so FNA for the demonstration network can be quantified. Under normal loading conditions, we did not observe any voltage violations or congested lines, thus no flexibility was needed.

4.1.3 Data-driven State Estimator (DdSE) - INESC

The Data-driven State Estimator (DdSE) provides the most likely state of the grid, comprising voltages and active powers, departing from a subset of meters that communicate in real time and supported by historical data. This means that the real-time monitoring of a grid is achieved without a heavy communications infrastructure and without replacing all meters by technologies capable of broadcasting their measurements in real time.

In the current implementation of this LV monitoring Demonstrator, meters are split as follows:

- 10 meters communicate voltage and active power every 30 minutes.
- 47 meters store their readings of voltage and active power and communicate them by the end of the day.

The arrival of a new set of real-time measurements (every 30m) triggers the DdSE to reconstruct a new state of the system. The voltage estimations for two connection points of the Demonstrator grid over 10 days are shown in Figure 4.15 and Figure 4.16. According to the results, it is evident that the tool is highly reliable in providing estimations that closely match the actual values. This degree of accuracy offers significant reassurance to the system operator, as it enhances its ability to make informed decisions about the system's operations.

Figure 4.15 - Comparison between estimated and real voltage values for one smart meter connected to phase a, over the period of 10 days

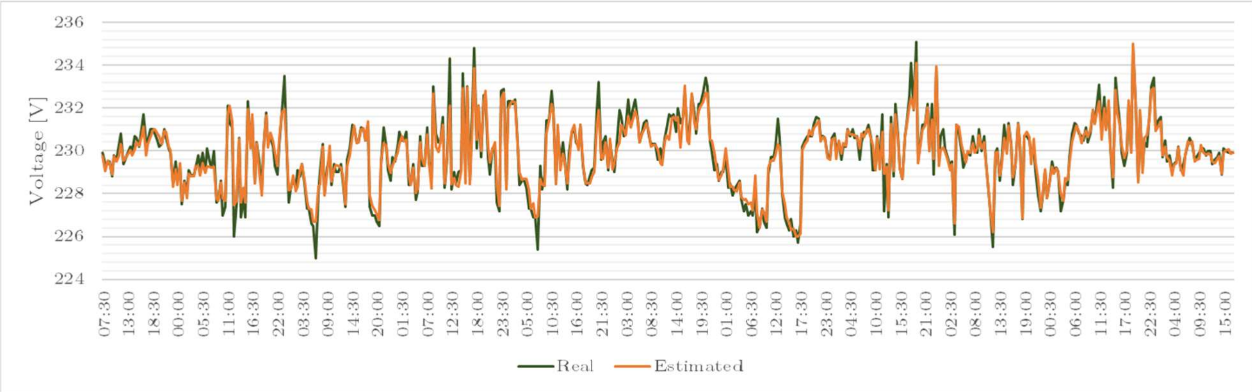


Figure 4.16 - Comparison between estimated and real voltage values for one smart meter connected to phase c, over the period of 10 days.

At the current stage of integration of the following metrics characterize the accuracy of the estimations when evaluating 10 days of estimations:

- Maximum absolute deviation (MAD): 4.0110 V
- Mean average error (MAE): 0.4817 V
- Maximum square error (MSE): 0.4530 V

As the amount of historical data grows in the coming months, the algorithm will have more data to analyze and find patterns that can help reconstruct the system state more accurately. This increase in data is expected to lead to improvements in the metrics used to evaluate the algorithm's performance. In addition, with a sufficiently large dataset, the algorithm will be able to estimate active power injections, further enhancing its capabilities.

4.2 Market-based flexibility service selection and activation

The market-based procurement process in the Demonstrator is set up according to the digital flexibility value chain, illustrated in Figure 4.17, with the DSO MITNETZ STROM as a buyer, Centrica as Flexibility Service Provider (FSP), NODES as independent market operator and the UMEI as standard communication interface to enable the connection of the DSOs to multiple market platforms. N-SIDE`s optimal bid recommender is interconnected on the buyer' side to ensure the selection of the most effective flexibility service for the specific grid problem as outlined in deliverable D8.1.

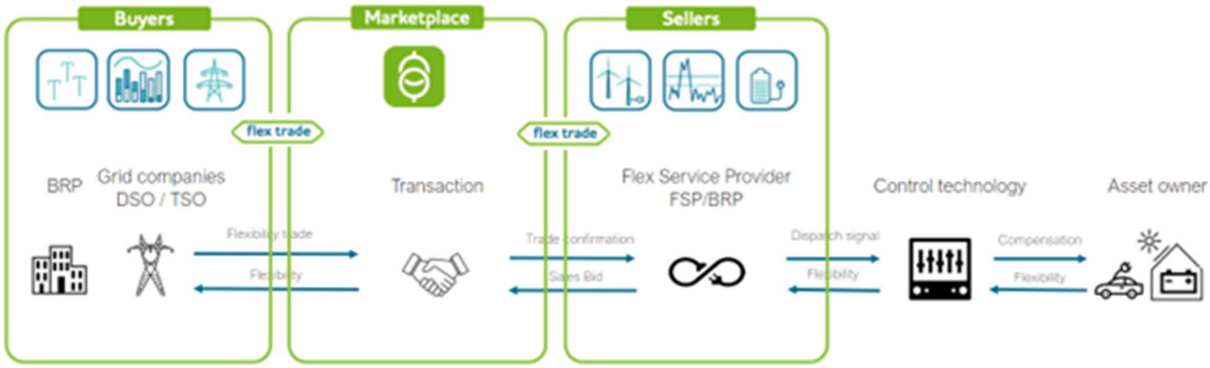


Figure 4.17 - Digital Flexibility value chain (www.nodesmarket.com)

4.2.1 Optimal Bid Recommender - N-SIDE

At regular intervals, the DSO will launch the N-SIDE Optimal Bid Recommender (OBR) tool that will analyse and identify the combination of bids that solves as many congestions as possible and at the lowest price. Based on the output of the Optimal Bid Recommender (OBR), the DSO will then be able to submit 'buy' orders that match the recommended 'sell' orders on the market platform.

At every call, the Optimal Bid Recommender (running on the servers of the DSO) will receive the needed inputs, which can be separated into three categories (Figure 4.18)

- The grid static data: macro view of the topology (i.e. which zones are connected)
- The grid state forecast (dynamic): headroom per transformer, headroom per feeder
- The market-related information: baseline of each flexible asset, flexibility bids (price and volume)

Once all the required information is gathered, a market-clearing engine will determine the optimal bids to select to resolve the identified grid congestions at minimum costs for the DSO. This process is based on a mathematical optimization problem.

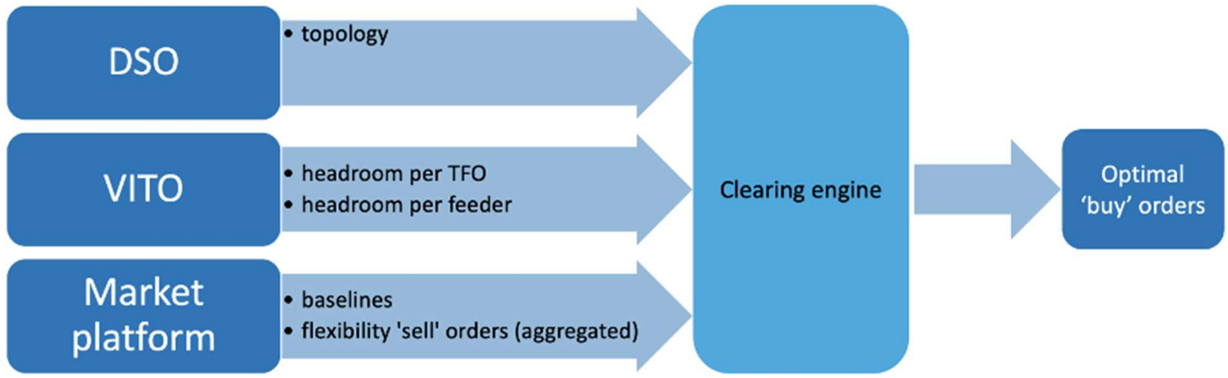


Figure 4.18 - Inputs and Outputs of the N-SIDE Optimal Bid Recommender

Once a solution is found, the bids to be selected are transmitted to the DSO. Then, the DSO will connect to the flexibility market via the UMEI to activate the sell offers by submitting a corresponding buy order. In case some congestions cannot be solved with the flexibility offers available on the market (i.e. not enough flexibility is offered), the DSO needs will anyway be able to leverage the N-SIDE tool by publishing the exact missing needs on the market platform through flexibility 'buy' offers. In that case, the DSO will choose the price of these offers beforehand without recommendation of the OBR, likely based on a limit price allowed by the regulation.

To ensure the tool is working correctly, we conducted several test sets. These tests included checks of individual functions of the algorithm (unitary tests) and verification of the entire tool's functionality with different input scenarios (functional/system tests). We used a test set of 30 scenarios with varying numbers of periods, headrooms and orders. The 30 scenarios passed according to all test success criteria defined, i.e. correct selections of bids, optimality of the solution proved and short sufficient computation time.

4.2.2 Flexibility service provider – Centrica

Centrica has the role of the FSP in the German Demonstrator and offers the flexibility services of one or multiple resource(s) through aggregation to system operators via a flexibility market. In this Demonstrator, flexibility should be provided for the DSO (MNS), while NODES is the Flexibility Market Operator (FMO).

The activities of FSP to provide flexibility for DSO in this Demonstrator can be divided into two categories:

•<u>Category 1: random activities</u>

- Registering flexible assets in the market platform and assigning them to a portfolio. This activity will be done in the beginning of the field test or will be repeated when a new flexible asset is available in one of the test grid zones of the Demonstrator.
- Developing the mathematical model for different type of flexible assets, which help afterward with the calculation of optimal bid.

•<u>Category 2: Regular activities</u>

Figure 4.19 Figure 4.19 illustrates different processes included in this category.

- Solving optimization to calculate optimal bid for each portfolio. The output of this optimization included the optimal flexibility volume together with its optimal cost to be offered via UMEI to FMO.
- Calculating the baseline for each portfolio and submit it to the FMO via the UMEI. Depending on the type of flexible asset and available historical data set, different methods can be used to calculate the baseline (as no standardized convention for baseline calculation is in place at this point).
- Receiving the trades information from the market via the UMEI and dispatching and disaggregating them to the flexible assets to deliver the activated flexibility. The Home energy management system (HEMS) will enable the interaction between Centrica's dispatch tool and the real flexible assets.
- Receiving sub-metering data via the HEMS to calculate flexibility and to monitor the real-time delivery of the flexibility.

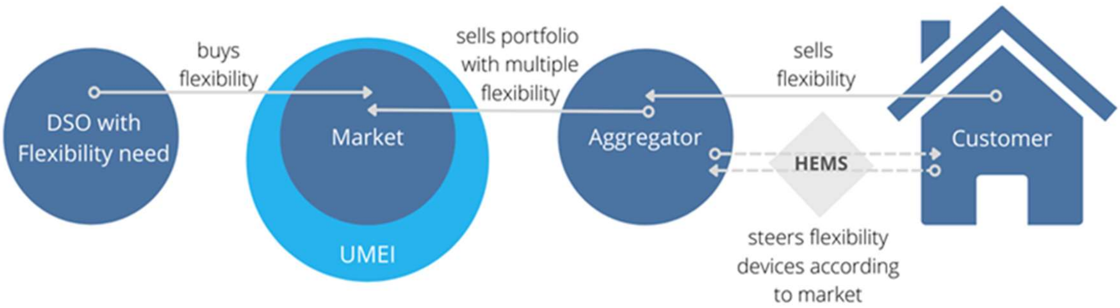


Figure 4.19 - Organization of data flows and asset steering in the German Demonstrator

4.2.3 Market-based flexibility procurement on NODES market platform via the UMEI

NODES as independent market operator provides the central environment for market-based procurement of flexibility in this Demonstrator ensuring correct and transparent transactions between buyers and sellers (without consideration of the post-trading activities, ie. validation and settlement, as these functionalities are not yet implemented in the UMEI V01. Using NODES integrated market design, shown in Figure 4.20, flexibility can be offered bottom-up and bought top-down allowing for an efficient use of available flexibility resources across all grid levels. Due to limited market liquidity at this point in time, NODES only applies a continuous market clearing via pay-as-bid.

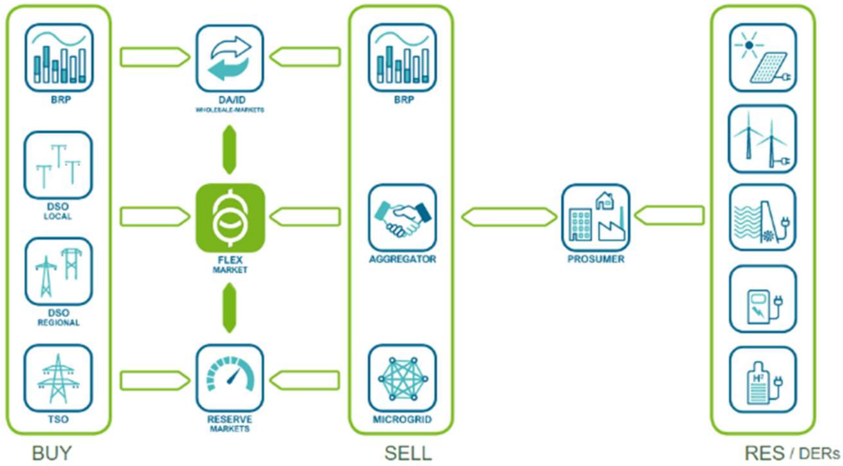


Figure 4.20 - NODES integrated market design (www.nodesmarket.com)

NODES flexibility market covers all necessary services related to each trading phase, i.e. registration and prequalification, flexibility procurement, and validation and settlement. Validation and settlement, however, will not be tested in the German Demonstrator because the major focus was setting up the operational flexibility value chain and the realization of market-based flexibility trading via the UMEI.

Due to the limited customer participation an initially defined KPI to reflect the market liquidity was not used as performance indicator.

4.2.3.1 Integration of NODES market platform and the UMEI

The functionalities required for the FMO and UMEI integration can be broadly divided into two categories:

- Category 1: Random operations: Registration & prequalification (grid data and nodes/asset data)
- Category 2: Regular/repetitive operations: Trading operations, baseline and meter readings registration and update

Due to the focus on the integration of the entire flexibility value chain for market-based flexibility procurement, and time constraints, only functionalities of category 2 have been implemented in the first version of the UMEI (Technical details about each operation and related functions are published in the UMEI API management and documentation (D2.4 - 2.6). Therefore, operations of category 1 have been performed directly via NODES graphical user interface or NODES API. The main operations of category 2 have been tested on the NODES market platform via the UMEI API:

- Communicating demand for flexibility: DSOs and other market participants communicate their demand for flexibility by posting buy orders. Each buy order is connected to a specific previously defined node in the grid.

- Communicating availability of flexibility: FSPs communicate their availability of flexibility by posting sell orders. Each sell order is connected to an asset portfolio. Every asset is assigned to a specific grid node. The UMEI will support posting buy orders, fetching the user-specific orders, and deleting (deactivating) orders.
- Registering baselines for asset portfolios that participate in the market: Baselines are planned/forecasted power consumption and/or production profiles without activation of flexibility of a portfolio that must be submitted by FSPs together with a sell order. The UMEI will support registering/fetching/deleting baselines.
- Registering meter readings for asset portfolios: Meter readings represent actual measured power consumption and/or production of a portfolio. However, submission/fetching of meter reading is not supported by the UMEI V01 and hence not part of the tested scenarios
- Receiving notifications: NODES market allows for the configuration of notifications of specific information such as upon a match of corresponding buy and sell orders, specific order submission etc. This option was also implemented in the UMEI.

4.2.3.2 Field test: Registration and prequalification

The operational steps categorized into of category 1 Registration and preregistration were conducted using NODES graphical user interface and API.

The system operator registers the grid license area on Nodes market platform and sets up the flexibility markets. The system operator thereby choses the granularity and amount of detail required to effectively activate flexibility from distributed assets to solve local or regional grid congestions. The FSP(s) register the assets in the respective license area providing the geographic coordinates of the assets, installed capacity or max. available flexibility and the meterpoint ID. As a final step the system operator confirms the registered assets by assigning them to a grid node. Following this protocol, the flexibility markets that were tested in the Mitnetz license area are represented on NODES flexibility market as shown in Figure 4.21. Each feeder is represented by a grid node (circle) where the size of each grid node represents the respective MV and LV feeders and the associated order book (to activate the flexibility of DER assets. Dashed lines indicate connected grid nodes allowing flexibility to be activated in connected grid nodes. Congested grid areas and the associated grid node are marked in red. Figure 4.22 shows a detailed view on the registered assets at LV grid node KV Nordstr. /Wurzener Str. 7260/MV feeder MFn4420.

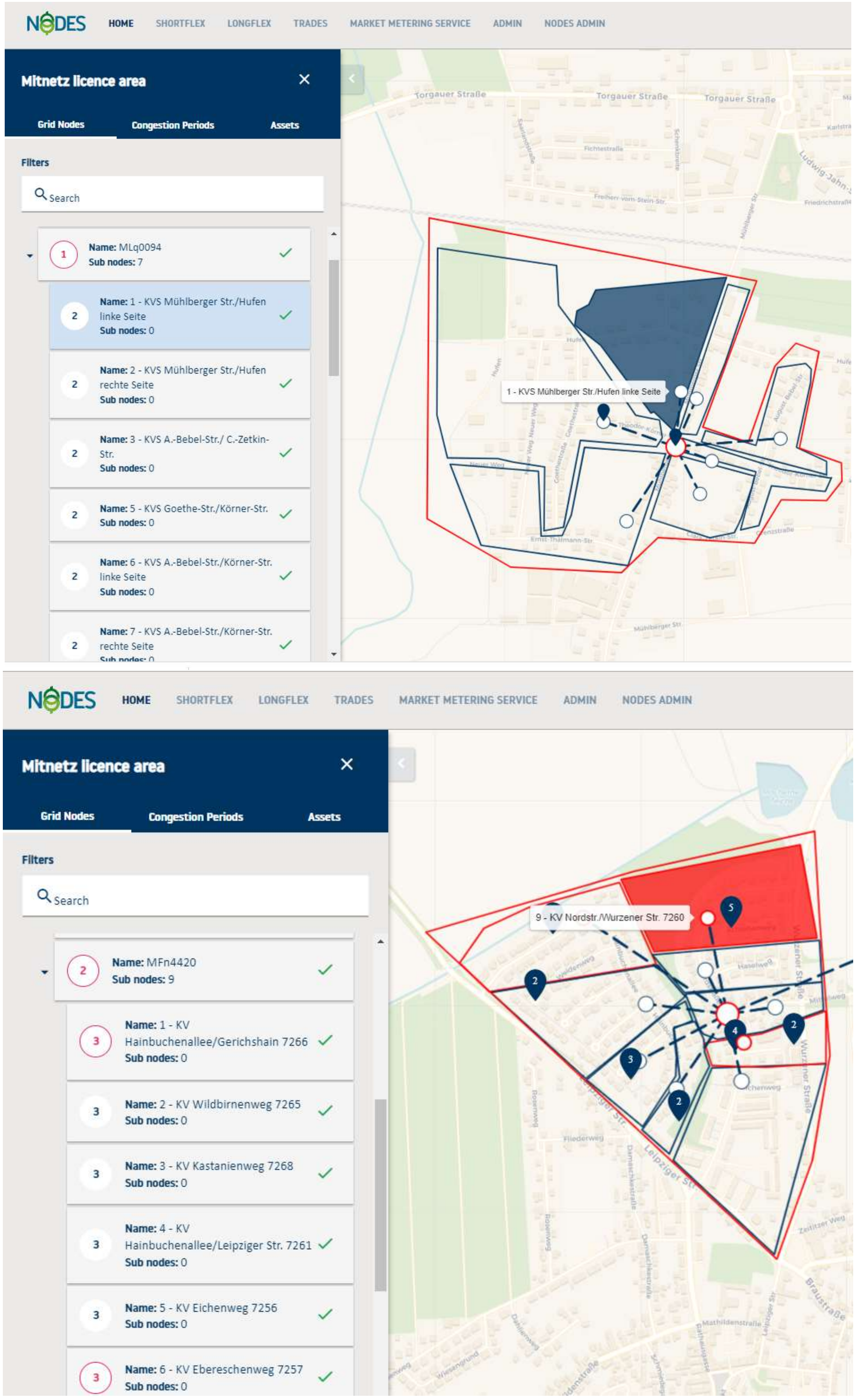


Figure 4.21 - Mitnetz license areas MV feeder MLq0094 (above) and MV feeder MFn4420 (below) and associated LV feeders on NODES market platform.

Figure 4.22 - Detailed representation of registered assets at LV grid node KV Nordstr. /Wurzener Str. 7260/MV feeder MFn4420

On the NODES market platform, a grid node corresponds to an order book (Figure 4.23). The flexibility procurement via the market platform is conducted in the order book that corresponds to the grid node where the congestion was identified.

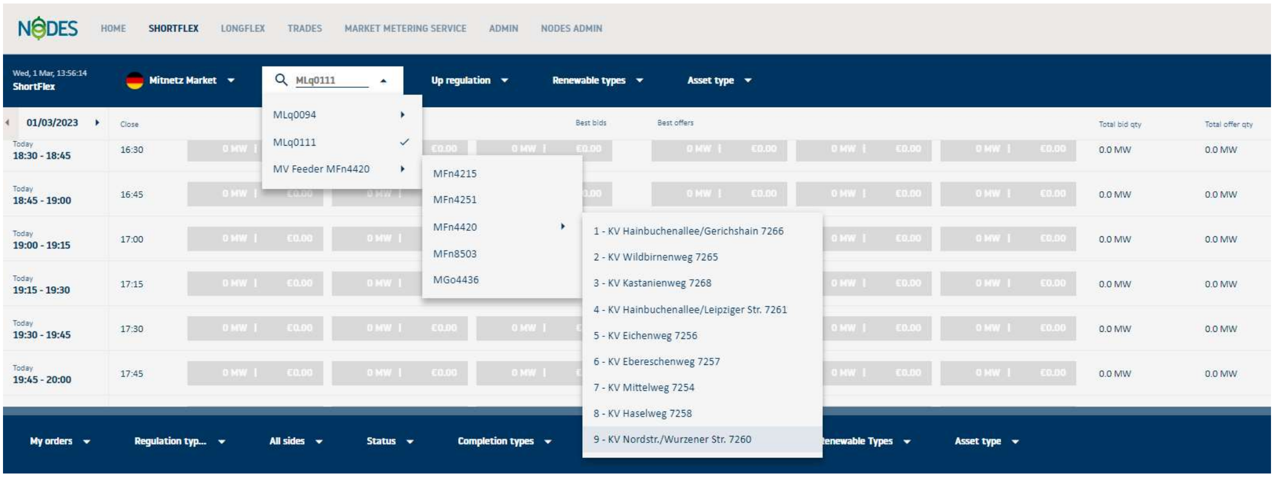


Figure 4.23 - Order book structure according to the grid node organization in the Mitnetz license area

4.2.3.3 Field test: Flexibility trading

The following operational steps form part of category 2 and are conducted via the UMEI. The UMEI API orders for every operation are presented in Tables 4.1 to 4.3

Two test series were performed involving Mitnetz, Centrica and NODES:

1) UMEI API In-Depth Functional Testing using Postman and the Swagger UI (Table 4.1 and Table 4.2); to ensure a) the correct integration of the UMEI with the NODES market platform and b) that all operations and related functions that a relevant for the actual trading phase are performing correctly;

2) Concrete trading scenarios between FSP (Centrica), FMO (NODES) and DSO (Mitnetz) to verify the correct market operation and matching while considering the common components to characterize the requested or offered flexibility, ie. location, volume and price (Table 4.3).

Test series 1:

Table 4.1 - Token-based authentication of the client

ID	**Function Name**	**HTTP Method**	**Endpoint**	**Request description**	**Response Code / Status**	**Response description**
A.1	Verify the authentication (request for OAuth2 access token)	POST	Authorization endpoint	Require access to the UMEI – initiate request for Access Token after client id and client secret are acquired.	**200 OK**	Authentication is successful - Access Token issued, access to resources is granted.
A.2	Verify the Access Token	GET, POST, DELETE, PATCH, PUT	Test API URLs	Request access to appropriate resources using Access Token	**200 OK**	Access Token is accepted and validated. The required resources are returned as a response.

Table 4.2 - Basic requests

ID	**Function Name**	**HTTP Method**	**Endpoint**
Portfolio			
P.1	Get all available asset portfolios	GET	/portfolios
P.2	Get an asset portfolio by Id	GET	/portfolios/{id}
P.3	Post a new DSO/FSP asset portfolio on the platform	POST	/portfolios
P.4	Partially update (updating particular fields) an existing DSO/FSP asset portfolio by Id	PATCH	/portfolios/{id}

P.5	To replace an existing DSO/FSP asset portfolio entirely by Id	PUT	/portfolios/{id}
P.6	Delete an existing DSO/FSP asset portfolio	DELETE	/portfolios/{id}
Baselines			
B.1	Post a new baseline interval for DSO/FSP asset portfolio	POST	/BaselineIntervals
B.2	Partially update an existing DSO/FSP baseline interval	PATCH	/BaselineIntervals/{id}
B.3	Replace an existing DSO/FSP baseline interval	UPDATE	/BaselineIntervals/{id}
B.4	Delete a DSO/FSP baseline interval from the platform by Id	DELETE	/BaselineIntervals/{id}
B.5	Get all portfolio baselines available for DSO/FSPs incl. FSPs/DSOs submitted	GET	/BaselineIntervals
B.6	Get a baseline interval by Id	GET	/BaselineIntervals/{id}
Market information			
M.1	Get all markets registered on the market platform	GET	/Markets
Orders			
O.1	Get all public orders on the market platform	GET	/PublicOrders
O.2	Get all existing orders on the market platform	GET	/Orders
O.3	Get an existing order by Id	GET	/Orders/{id}
O.4	Submit a new buy/sell order by DSO/FSP	POST	/Orders
O.6	Update an existing order from the market platform	PUT	/Orders/{id}
O.7	Delete an existing order from the market platform	DELETE	/Orders/{id}
Trades			
T.1	Get all available trades on the platform	GET	/Trades
Flexibility zone			
F.1	Post a new flexibility zone	POST	/FlexibilityZones
F.2	Get all flexibility zones on the market platform	GET	/FlexibilityZones
Meter Readings			
MR.1	Post a new meter reading	POST	/MeterReadings
MR.2	Get all meter readings	GET	/MeterReadings

Task F.1 and F.2 were not tested in the German Demonstrator and instead inserted manually via NODES UI. The tasks MR.1 and MR.2 were not tested as the data at the point of connection to the grid will be provided by HEMS directly and Smart Meter rollout in the grid area of the German Demonstrator is scarce.

On NODES market platform every grid node corresponds to an orderbook. As such the DSO can procure the flexibility according to the inserted grid structures (Figure 4.24).

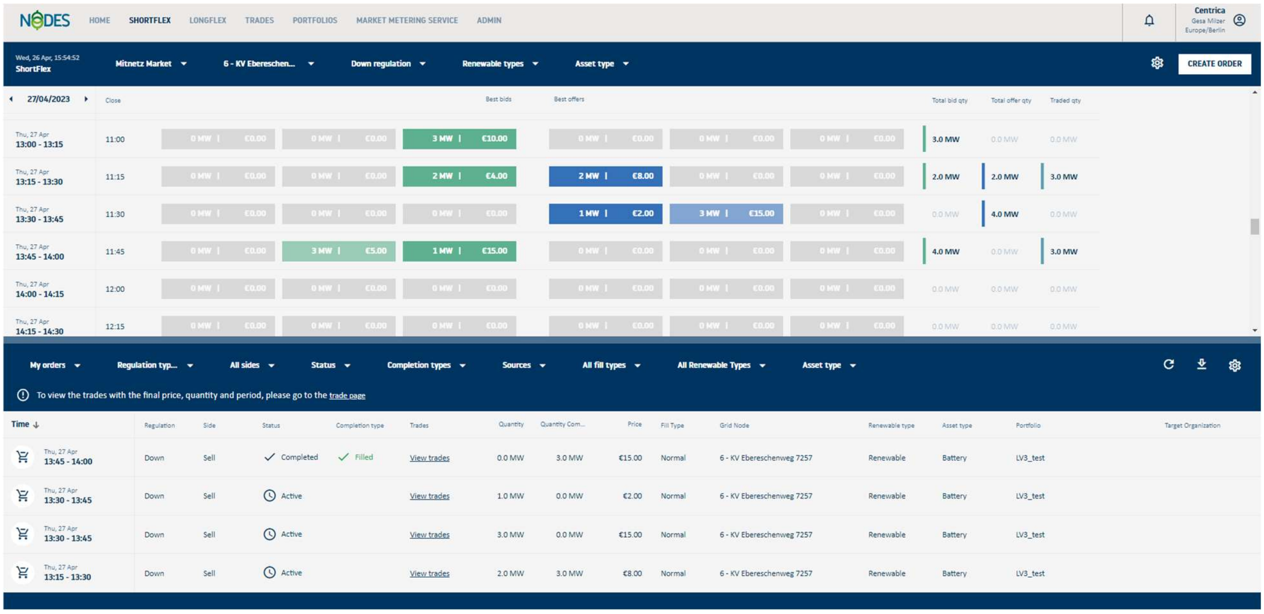


Figure 4.24 - NODES Market Platform: Trading in MFn4420 Orderbook

Table 4.3 provides an overview of the testes market scenarios:

- Scenarios A+B cover the submission of order by all market participants in different order books considering different flexibility volumes and the direction of the regulated flexibility (up/down regulation).
- Scenario C was performed as a consequence of the scenarios A+B ensuring the correct submission of baselines for each sell order that has been created in scenarios A+B. The baselines are accessible as csv files.
- Scenario E was tested to prevent incorrect matching of orders.
- Scenario F was performed to ensure the correct purchase of flexibility across different grid levels via connected order books.
- Scenarios G serve to facilitate the operational routine for the DSO to get an overview of the flexibility offers and to feed the information into the optimal bid recommender to select the optimal offer for the respective grid problem.
- The notification function on NODES platform (Scenario H) was tested to configure the messaging service according to the needs of each market participant.
- Ultimately, all transactions were exported in Scenario J

Table 4.3 - Trading scenarios

Scenario	Scenario	Role	Action	Grid Area	Time (CET)	Quantity [MW]	Direction	Price [€/MW]	UMEI API function
Scenario A)	Offer FSP, need DSO on same grid area								
A.1		FSP	Sell Order	MLq0094	17.06.2022 10am-2pm	3	down	10	POST /Orders
		DSO	Match Order	MLq0094	17.06.2022 10am-2pm	3	down	10	POST /Orders
A.2		FSP	Sell Order	1 - KV Hainbuchenallee/Gerichshain	17.06.2022 5pm-7pm	2	up	8	POST /Orders
		DSO	Match Order	1 - KV Hainbuchenallee/Gerichshain	17.06.2022 5pm-7pm	2	up	8	POST /Orders
A.3		FSP	Sell Order	9 - KV Nordstr./Wurzener Str. 7260	17.06.2022 6am-7am	5	up	9	POST /Orders
		DSO	Match Order	9 - KV Nordstr./Wurzener Str. 7260	17.06.2022 6am-7am	5	up	9	POST /Orders
Scenario B)	Offer DSO, need FSP on same grid area								
B.1		DSO	Buy Order	MFn4420	18.06.2022 10am-2pm	3	down	10	POST /Orders
		FSP	Match Order	MFn4420	18.06.2022 10am-2pm	3	down	10	POST /Orders
B.2		DSO	Buy Order	1 - KV Hainbuchenallee/Gerichshain	18.06.2022 5pm-7pm	2	up	8	POST /Orders
		FSP	Match Order	1 - KV Hainbuchenallee/Gerichshain	18.06.2022 5pm-7pm	2	up	8	POST /Orders
B.3		DSO	Buy Order	MV Feeder MFn4420	18.06.2022 6am-7am	5	up	9	POST /Orders
		FSP	Match Order	MV Feeder MFn4420	18.06.2022 6am-7am	5	up	9	POST /Orders
Scenario C)	Baselining								
C.1		FSP	Send Baselines	MLq0094	19.06.2022 10am-2pm				POST /BaselineIntervals
		DSO	Get Baselines	MLq0094	19.06.2022 10am-2pm				GET /BaselineIntervals/{id}
C.2		FSP	Send Baselines	1 - KV Hainbuchenallee/Gerichshain	19.06.2022 5pm-7pm				POST /BaselineIntervals
		DSO	Get Baselines	1 - KV Hainbuchenallee/Gerichshain	19.06.2022 5pm-7pm				GET /BaselineIntervals/{id}
C.3		FSP	Send Baselines	9 - KV Nordstr./Wurzener Str. 7260	19.06.2022 6am-7am				POST /BaselineIntervals
		DSO	Get Baselines	9 - KV Nordstr./Wurzener Str. 7260	19.06.2022 6am-7am				GET /BaselineIntervals/{id}
Scenario D)	Offer DSO, Counteroffer FSP								
D.1		DSO	Buy Order	MFn4420	20.06.2022 10am-2pm	3	down	10	POST /Orders
		FSP	Sell Order	MFn4420	20.06.2022 10am-2pm	3	down	13	POST /Orders
D.2		DSO	Buy Order	1 - KV Hainbuchenallee/Gerichshain	20.06.2022 5pm-7pm	2	up	8	POST /Orders
		FSP	Sell Order	1 - KV Hainbuchenallee/Gerichshain	20.06.2022 5pm-7pm	1	up	8	POST /Orders
D.3		DSO	Buy Order	MV Feeder MFn4420	20.06.2022 6am-7am	5	up	9	POST /Orders
		FSP	Sell Order	MV Feeder MFn4420	20.06.2022 6am-7am	8	up	9	POST /Orders
Scenario E)	Offer FSP, Counteroffer DSO								
E.1		FSP	Sell Order	MLq0094	21.06.2022 10am-2pm	3	down	10	POST /Orders
		DSO	Buy Order	MLq0094	21.06.2022 10am-2pm	3	down	13	POST /Orders
E.2		FSP	Sell Order	1 - KV Hainbuchenallee/Gerichshain	21.06.2022 5pm-7pm	2	up	8	POST /Orders
		DSO	Buy Order	1 - KV Hainbuchenallee/Gerichshain	21.06.2022 5pm-7pm	1	up	8	POST /Orders
E.3		FSP	Sell Order	9 - KV Nordstr./Wurzener Str. 7260	21.06.2022 6am-7am	5	up	9	POST /Orders
		DSO	Buy Order	9 - KV Nordstr./Wurzener Str. 7260	21.06.2022 6am-7am	8	up	9	POST /Orders
Scenario F)	DSO MV need, FSP LV Offers								
F.1		FSP	Sell Order	6 - KV Ebereschenweg	22.06.2022 10am-2pm	3	down	10	POST /Orders
		DSO	Buy Order	MFn4420	22.06.2022 10am-2pm	20	down	10	POST /Orders
F.2		FSP	Sell Order	6 - KV Ebereschenweg	22.06.2022 5pm-7pm	2	up	10	POST /Orders
		DSO	Buy Order	MV Feeder MFn4420	22.06.2022 5pm-7pm	15	up	8	POST /Orders
F.3		FSP	Sell Order	5 - KVS Goethe-Str./Körner-Str.	22.06.2022 6am-7am	5	up	9	POST /Orders
		DSO	Buy Order	MLq0094	22.06.2022 6am-7am	25	up	9	POST /Orders
Scenario G)	Export all bids from Market								
G.1		FSP	Send Bids	MLq0094	19.06.2022 10am-2pm	-	-	-	POST /Orders
		DSO	Get Bids	MLq0094	19.06.2022 10am-2pm	-	-	-	GET /Orders
G.2		FSP	Send Bids	1 - KV Hainbuchenallee/Gerichshain	19.06.2022 5pm-7pm	-	-	-	POST /Orders
		DSO	Get Bids	1 - KV Hainbuchenallee/Gerichshain	19.06.2022 5pm-7pm	-	-	-	GET /Orders
G.3		FSP	Send Bids	9 - KV Nordstr./Wurzener Str. 7260	19.06.2022 6am-7am	-	-	-	POST /Orders
		DSO	Get Bids	9 - KV Nordstr./Wurzener Str. 7260	19.06.2022 6am-7am	-	-	-	GET /Orders

5 Conclusion and Outlook

This report documents the technical and operational integration of each tool and market component into the Mitnetz system to ensure a reliable and secure data exchange and interoperability between the systems. Furthermore, each smart grid tool, test set and result are presented to explain the objective and functioning of each tool for the purpose of congestion detection and flexibility need assessment to prove the reliability and accuracy of each tool considering the available data and data quality. The market participants interacting in the framework of market-based flexibility procurement presented their tools and platforms that are designed to facilitate the access to DERs for DSOs while aiming for an overall optimization in terms of use of resources, bid selection and cost minimization. Specific scenarios are tested to ensure the operational reliability between the systems via the UMEI, NODES UI and NODES API and to simulate trading scenarios.

The conducted test sets per tool and in the market environment successfully accomplished the initially defined MVPs, proving the usability within the digital flexibility value chain despite of limited data availability and partly regular quality of data sets to achieve the desired accuracy. Furthermore, the low number of participants and as such more diverse prosumer asset types and load profiles impedes the replication of more realistic scenarios in terms of grid load patterns, potential congestions and thus market situations in the tested grid areas.

This scarcity of data may be reduced through the progress of the Smart meter roll-out in Germany and enhanced, simplified, and standardized installation of HEMS and BEMS. Furthermore, current regulatory barriers affecting the implementation of flexibility markets such as grid tariff structure, taxation scheme as well as flexibility asset categorization disincentivizes industrial and residential prosumers to offer their flexibility for grid management.

The next step of the German Demonstrator will be the examination of the entire digital flexibility value chain as a combined process of all tools and market components. The objective is to evaluate the tool and system performance as well as numerical results using the common EUniversal project KPIs and the KPIs defined for the German Demonstrator. The results will be described in detail in D8.3.

6 References

[1] M. U. Hashmi, A. Koirala, R. Lundholm, H. Ergun and D. V. Hertem, “Evaluation of voltage magnitude-based unbalance metric for low voltage distribution networks,” in *IEEE Power & Energy Society General Meeting (PESGM)*, Denver, Colorado, USA, 2022.

[2] M. U. Hashmi, D. Brummund, R. Lundholm, A. Koirala and D. V. Hertem, “Consensus based phase connectivity identification for distribution network with limited observability,” *Sustainable Energy, Grids and Networks,* 2023.

[3] M. U. Hashmi, A. Koirala, H. Ergun and D. V. Hertem, “Robust flexibility needs assessment with bid matching framework for distribution network operators,” *Sustainable Energy, Grids and Networks,* 2023.

[4] M. U. Hashmi, A. Koirala, H. Ergun and D. V. Hertem, “Flexible and curtailable resource activation in a distribution network using nodal sensitivities,” in *International conference on smart energy systems and technologies (SEST)*, Finland, 2021.

[5] M. U. Hashmi, A. Koirala, H. Ergun and D. V. Hertem, “Flexible and curtailable resource activation in three-phase unbalanced distribution networks,” *Electric Power Systems Research,* 2022.

[6] M. U. Hashmi, A. Koirala, H. Ergun and D. V. Hertem, “Perspectives on distribution network flexible and curtailable resource activation and needs assessment,” *IEEE Transactions on Industry Applications,* 2023.

[7] M. U. Hashmi, A. Koirala, H. Ergun and D. V. Hertem, “Flexibility needs assessment for distribution networks,” in *Risk Days*, London, 2023.

[8] C. Hauk, „Systemstudie Flexibilität Projektangebot für E.ON SE im Rahmen der SDES Roadmap,“ RWTH Aachen University, 2022.

7 Annex

Table 7.1 - Overview internal data exchange

Phase	Step	Periodicity	Operation description	Requester tool	Replier tool	Request			Response		
						HTTP method	Parameters / Body	Format	Status	Content / Body	Format
I	1.1.1	every 24 hours	Push historical data	Synchronizer	State Estimator	POST	Historical data	JSON	200 OK	-	-
		every 15 min	Push real measurements	Synchronizer	State Estimator	POST	Real measurements	JSON	200 OK	-	-
	1.1.2	every 15 min	Initiate state estimator	Synchronizer	State Estimator	GET	Network ID	JSON	200 OK	-	-
	1.1.3		Request real-time state estimation results	Synchronizer	State Estimator	GET	Network ID	JSON	200 OK	Most recent snapshot of the network, optimal power flow (OPF) calculated	JSON
	1.2.1	every 15 min	Push weather forecasts	Synchronizer	Congestion Forecasting & Prevention Tool	POST	Weather forecasts	JSON	200 OK	-	-
			Push grid measurements	Synchronizer	Congestion Forecasting & Prevention Tool	POST	Grid measurements	JSON	200 OK	-	-
	1.2.2	every 15 min	Trigger the calculation	Synchronizer	Congestion Forecasting & Prevention Tool	GET	List of cases to be calculated	JSON	200 OK	-	-
	1.2.3	as soon as ready	Push calculation results	Congestion Forecasting & Prevention Tool	Synchronizer	POST	List of calculated results: headrooms for the specified list of cases	JSON	200 OK	-	-
II	2.1.1	every 15 min	Push grid layout	Synchronizer	Flexibility Needs Assessor	POST	The updated version of grid layout	JSON	200 OK	-	-
			Push optimal power flow & generic load profiles	Synchronizer	Flexibility Needs Assessor	POST	OPF and generic load profiles provided by State Estimator (step 1.1.3)	JSON	200 OK	-	-
		every 24 hours	Push Annual Load Consumption	Synchronizer	Flexibility Needs Assessor	POST	Load consumption for the previous year	JSON	200 OK	-	-
		every 15 min	Push nodal day-ahead forecast of P and Q	Synchronizer	Flexibility Needs Assessor	POST	Nodal day-ahead forecast of P and Q	JSON	200 OK	-	-
	2.1.2	every 15 min	Initiate flex needs assessment	Synchronizer	Flexibility Needs Assessor	GET	t.b.d.	JSON	200 OK	-	-
	2.1.3	as soon as ready	Retrieve the results of flex needs assessment	Synchronizer	Flexibility Needs Assessor	GET	t.b.d.	JSON	200 OK	Nodal and zonal flexibility needs assessment	JSON
	2.2.1	every 15 min	Push updated grid hierarchy, sell-orders, baselines and headrooms + start the calculation session	Synchronizer	Optimal Bid Recommender	POST	Updated grid hierarchy, sell-orders, baselines and headrooms (step 1.2.3)	JSON	200 OK	Session ID	JSON
	2.2.2	as soon as ready	Retrieve calculation results for the session	Synchronizer	Optimal Bid Recommender	GET	Session ID	JSON	200 OK	Activation Cost, Power Accepted in Orders, Congested Grid Zones	JSON