AI·EFFECT

Artificial Intelligence Experimentation Facility
For the Energy seCTor

D1.1 - Design of AI-EFFECT Use Cases and Outputs

Funded by the European Union

AI-EFFECT is supported by the European Union's Horizon Europe programme under agreement 101172952. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Document Information

DOCUMENT/DELIVERABLE ID	D1.1
TYPE	Report
DISTRIBUTION LEVEL	Public
DUE DELIVERY DATE	30.09.2025
DATE OF DELIVERY	
VERSION	V1.0
DELIVERABLE RESPONSIBLE	TU Delft
AUTHORS	Paul Bannmüller (TU Delft), Carlos Silva (INESC TEC), Johanna Vorwerk (DTU), Tobias Kasper Skovborg Ritschel (DTU), Ahmed Abdelgawad (Fraunhofer FIT), Gil Sampaio (INESC TEC), Luís Moncaixa (Watt-IS)
OFFICIAL REVIEWERS	RWTH, INESC TEC

Document History

VERSIONS	AUTHORS	DATE	CONTENT AND CHANGES
V0.1	Paul Bannmüller	01/09/2025	First Draft
V0.2	Carlos Silva, Leonardo Carreras	12/09/2025	First Draft Review
V0.3	Paul Bannmüller	19/09/2025	Revision of the whole report based on comments of the reviewers
V0.4	Carlos Silva, Leonardo Carreras	22/09/2025	Second Review
V1	Paul Bannmüller	23/09/2025	Final Version

Contents

List of Figures

List of Tables

Executive Summary

Work Package WP1, and specifically Task T1.1, establishes the foundation of the AI-EFFECT project. Its primary objective is to design the nodes and **TEF Services** from which the fundamental structure of the TEF and requirements for the platform are developed and refined in subsequent work packages WP2 and WP3. The report frames each **TEF Test Use Case** as a pipeline of modular TEF Services applied to a **Business Use Case**, including services from data provision and synthetization to explainability, mathematical verification, benchmarking, simulation/lab testing, and human-AI interaction.

There are two **main outcomes** of T1.1: First, the definition of the TEF Services, which pose a minimal set of services that will be realized within the AI-EFFECT project. Second, the definition of the TEF Test Use Cases and the corresponding Business Use Cases, documented the TEF Test Use Case templates based on the HE AI4REALNET MS Word template adapted from ISO/IEC TR 24030.

Four nodes host at least one TEF Test Use Case: The **Danish node** focuses on multi-energy and sector coupling with district heating, offering a full validation pipeline including explainability analysis, mathematical verification, lab-based testing and living-lab testing to ensure that AI tools are robust, trustworthy and effective. The **German node** aims at DSO congestion management with benchmarking, explainability, verification, and testing on a simulated distribution grid. The **Dutch node** builds a testing workflow for transmission system operator congestion management tools with synthetic data generation, benchmarking, and human-AI-interaction testing. The **Portuguese node** hosts several TEF Test Use Cases around the topic of energy-community-scale efficiency and flexibility, covering virtual energy management, energy-sharing optimization, distributed energy resources (DER) control/scheduling, forecasting, and network monitoring across simulation based and hardware-based testing setups.

Analyzing the data flows of the preliminary node designs, a generalized data flow was derived. The included stakeholders at a generic node are the **data provider**, eventually complemented by intermediate data handling through data spaces, the **node host**, hosting virtual and/or physical labs, as well as the **AI vendor**, aiming to test its AI tool. While the AI vendor is often referred to as the main user, other stakeholders should also be entitled to use the TEF Services.

Within the TEF, there are three levels of data sharing: **Internodal** (highest level), **intranodal** (middle level) and **intra-stakeholder** (lowest level). For several use cases, sensitive data needs to be shared. Therefore, techniques such as data synthetization, aggregation, and anonymization can be applied to transform sensitive data into publicly shareable data. If it cannot be transformed into public data, techniques such as encryption, co-simulation or containerization may be used, and legal frameworks, such as non-disclosure agreements (NDAs) need to be established. **Interoperability** must be maintained throughout the platform on a functional level, on a data sensitivity level, on a syntactic, semantic, and technical level.

While the exact role and design of the digital **platform** complementing the TEF is still subject of discussion between WP1, WP2 and WP3, some core functionalities needed out of the view of WP1 are contributed to the discussion. This includes providing access to TEF users, enabling the utilization of TEF Services, enabling necessary data sharing, and reporting the results of the TEF Services back to the user. Two platform **scenarios** on a data flow level are introduced, presenting two extremes: the “lean platform” covering only interactions between TEF users and the platform, and the “holistic platform” covering data flows on all levels. In the next step, to progress the discussion, detailed **workflow descriptions** of the TEF Services will be developed and presented in the subsequent deliverables.

1 Introduction

The objective of Task 1.1 is to design the concrete use cases to demonstrate the unique functionalities of the AI-EFFECT testing and experimentation facility (TEF) and the platform; this includes designing the required interfaces with WP2 and WP3, the data flow, and the actors involved.

For the use cases, the report distinguishes between the **Business Use Case**, describing the application of an AI tool in the energy sector, and the **TEF Test Use Case**. The TEF Test Use Cases aim to test the Business Use Case and, therefore, are use cases from the perspective of the testing and experimentation facility. In this project, each of the four nodes hosts one or more TEF Test Use Cases. The four nodes are intended to showcase the capabilities of the AI-EFFECT platform and support its development. When the platform is deployed, new nodes can be added to cover different TEF Test Use Cases. Each TEF Test Use Case functions as a pipeline of **TEF Services** tailored to the Business Use Case. For example, this pipeline might include data provision, experimental design for lab testing, lab testing, explainability analysis, and human-AI interaction testing.

The purpose of this document is to define the use cases and TEF Services that demonstrate the functionalities of the AI-EFFECT platform. For this, a complete and detailed list of functionalities and specifications of the TEF has to be defined. This is done by defining the TEF Services in Section 2. After that, the specifications for the use cases that will demonstrate the key services offered by the AI-EFFECT TEF are identified. Therefore, each node and the corresponding TEF Test Use Cases are introduced, also in Section 2. Section 3 analyzes the use cases and node designs stated before to provide insights from work package WP1 that are important for developing the TEF and the digital platform in WP2 and WP3. Finally, Section 44 explains how the content of this report is relevant to other tasks and work packages within the AI-EFFECT project, and presents conclusions. The list of TEF Services, the detailed description of the TEF Test Use Case, as well as the template used for this can be found in the Appendix.

It is important to note that when writing this document, the project partners were still developing the nodes. Therefore, the design of the nodes described in this document could still change and should not be understood as promises.

Table 1: List of AI-EFFECT partners

Number	Legal Name	Acronym	Country
01	EPRI Europe DAC (Coordinator)	EEU	IE
02	Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência	INESC TEC	PT
03	Danmarks Tekniske Universitet	DTU	DK
04	Technische Universiteit Delft	TU Delft	NL
05	Institut de Recherche Technologique SystemX	IRTSX	FR
06	Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e. V.	Fraunhofer	DE
07	Rheinisch-Westfaelische Technische Hochschule Aachen	RWTH	DE
08	IKIM LTD.	IKIM	IE
09	Maynooth University	NUIM	IE
10	DNV AS	DNV	NO

11	EnliteAI GmBH	ENLITE	AT
12	Watt-IS S.A.	WATT-IS	PT
13	Cooperativa Eléctrica do Vale D'Este CRL	CEVE	PT
14	Bornholms Varme A/S	BHV	DK
15	ENEL GRIDS S.R.L	ENEL	IT
16	E Distribucion redes Digitales (Affiliated Entity)	EDRD	ES
17	Hertie School Gemmeinnutzige GMBH	HERTIE	DE
18	Center Danmark Drift APS	CDK	DK
19	Tennet TSO BV (Associated Partner)	TENNET	NL

2 TEF Test Use Cases

In this Section, the functional structure of the TEF is described in detail. For this, the TEF Services were defined and listed. An overview of the TEF Services is provided in Section 2.1. Furthermore, a detailed description of the use case goals and scenarios, as well as non-functional requirements was accomplished by defining the TEF Test Use Cases of each node through filling out the TEF Test Use Case template, one template for each TEF Test Use Case. The process of creating the templates is described in Section 2.2. The filled templates can be found in Section 5.2 in the Appendix, while on overview over each node and the corresponding TEF Test Use Cases is provided in Sections 2.32.3 -2.6.

2.1 Overview of TEF Services

TEF Services describe the core functionalities provided by the nodes via the AI-EFFECT platform. To support a more modular implementation and usage across the TEF, the TEF Services are defined independently of the use case. As described above, a TEF Test Use Case includes a pipeline out of several TEF Services applied to a Business Use Case. The list of TEF Services, shown in Table 3 in the Appendix, was derived by aggregating all important services of the four nodes, while aligning with the functionalities stated in the grant agreement. The TEF Services can be grouped into data provision and augmentation, data synthetization, AI tool verification and explainability analysis, and AI tool testing and benchmarking. It should be noted that the nodes will develop most of the TEF Services by the nodes for specific use cases, and, therefore, cannot be applied to other use cases without adaptations. The aim is to minimize the overhead for such adaptations.

Each node of the TEF hosts at least one TEF Test Use Case, consisting of several TEF Services. Table 2 maps the TEF Services to the nodes, with priorities assigned to the TEF Services of each node; the number one represents the highest priority. Several numbers for the same TEF Service at the Portuguese node refer to several TEF Test Use Cases, where the services are applied. It should be noted that while these TEF Services define the minimal viable product delivered at the end of the project, the nodes may add more services to the TEF during the project.

Table 2: TEF Service to node distribution, with priorities assigned.

#	TEF Service	Danish Node	Dutch Node	Portuguese Node	German Node
1	Data provision				2
2	Knowledge Store			2	
3	Data synthetization		1		
4	Benchmarking		2	4, 8, 11	1
5	Explainability Analysis	1		6	3
6	Environment for Mathematical Verification	2		9	4
7	Mathematical Verification	2		9	
8	Simulation-based Testing			1, 3, 7	

9	Experimental Design for Simulation-based testing			1, 3, 7	
10	Lab-based Testing	3		5, 12	
11	Experimental Design for Lab-based testing	3		5, 12	
12	Living-Lab Testing	4		10, 13	
13	Human-AI Interaction Testing		3		
14	Experimental Design for Human-AI Interaction Testing		3		

2.2 Creating the TEF Test Use Case Template

The TEF Test Use Case Template was developed based on the HE AI4REALNET MS Word template adapted from ISO/IEC TR 24030 [1], according to the standard methodology IEC 62559. The following changes have been applied to the AI4REALNET MS Word template:

- Adapting the template to the functional TEF structure: TEF Test Use Case, Business Use Case and TEF Services.
- Adding a detailed description of the TEF Test Use Cases in the form of KPIs, hypotheses, features, challenges and mitigation, future scope and others in Section 2 of the template.
- Adding a third section that focuses on the data flow, the AI tools under testing, the data synthetization and technical requirements for the execution of the testing, including the possible integration of VILLASnode [2] [3].
- Removing the sections on AI tool scenarios and the detailed information exchanged. This is covered in low detail in Section 3 of the template and will be part of the detailed workflow descriptions for each TEF Service.

It should be noted that the third section could not be filled out in detail by some of the nodes yet, as some of the details about the platform and the implementation of the nodes are still to be determined in WP2 and WP3. The missing information will be identified during discussions with WP2 and WP3 and addressed in the detailed workflow descriptions for each TEF Service, which will be included later deliverables. The TEF Test Use Case template can be found in the Appendix 5.2.

2.3 Danish Node

The Danish node of AI-EFFECT is located at the Technical University of Denmark (DTU), with activities distributed across the DTU Wind and DTU Compute Departments. It integrates both physical and digital infrastructure, enabling a hybrid testing environment that combines real-world data with controlled experiments.

The node supports a single TEF Test Use Case focused on multi-energy and sector coupling, specifically enhancing the coupled operation of district heating systems and distribution systems through AI. It provides a full validation pipeline including explainability analysis, mathematical verification, lab-based testing and living-lab testing to ensure that AI tools are robust, trustworthy and effective. This integrated approach aligns with the core functionalities of AI-EFFECT to provide an end-to-end certification pipeline from simulation to living labs, including the verification of AI tool coherency across space and time. In addition, the Danish node aligns with the AI-EFFECT goal of building a European infrastructure for AI experimentation offering modular services and realistic environments that support innovation in sustainable energy systems.

The narrative of the TEF Test Use Case: Multi-energy and sector coupling

The Danish node's TEF Test Use Case focuses on enhancing the operation of multi-energy systems, particularly the coupling between district heating and power systems. The use case supports AI developers in validating and improving their tools through a structured end-to-end testing pipeline that integrates four key TEF Services:

- **Explainability Analysis**: AI Models are analyzed to ensure that their predictions are explainable and interpretable, using techniques like Shapley values to assess how much features impact the predictions.
- **Mathematical Verification:** AI Tools are rigorously tested against defined constraints to obtain mathematical performance guarantees or detect unrealistic outputs.
- **Laboratory-Based Testing:** Conducted at DTU's SYSLAB, this service allows AI tools to be tested in a controlled physical environment simulating real multi-energy systems.
- **Living-Lab Testing:** AI tools are evaluated using real-world data from the energy grid operators BEOF and TREFOR, enabling a comparison of predicted setpoints with actual system behavior.

Together, these services form a comprehensive validation chain that ensures AI solutions are robust, explainable and effective under realistic operating conditions. The setup combines real-world data, physical infrastructure, and advanced analytics to support the development of reliable AI tools tailored to practical energy system challenges.

Corresponding Business Use Case(s)

The Danish node develops multiple business use cases to demonstrate the effectiveness of the TEF Test Use Case. These use cases are grouped into three thematic areas—**Forecasting**, **Control**, and **Operation**—and vary in both priority and the extent to which they are integrated into the TEF validation pipeline. Four business use cases have been selected to be demonstrated during the AI-EFFECT project, and additional ones have been identified for future exploitation and demonstration of the node's capabilities.

Forecasting use cases: These use cases focus on predicting energy demand and generation to support proactive and efficient system operation. All these use cases benefit from shared infrastructure and data services.

- **Spatial Hierarchies for District Heating Load Forecasting** *(highest priority)* Predicts heat demand across different subnets using time-series and graph-based models, enabling spatially granular control strategies.

- **District Heating Load Forecasting** *(highest priority):* Uses ARMAX models and weather data to forecast heat consumption, improving planning and reducing operational uncertainty.

- **Generation Forecasting** *(highest priority):* Predicts electricity production from solar PV installations using weather data, supporting integration of renewables into heating systems.

While explainability analysis and mathematical verification will be performed for all forecasting use cases, living-lab testing is only conducted for district heating load forecasting due to data availability. Lab-based testing will only be pursued if node partners identify a clear advantage over using real-world data. These forecasting use cases are foundational for enabling smarter control and optimization strategies and may be expanded into full validation pipelines in future iterations.

Control Use Cases: These use cases apply AI to dynamically adjust system parameters.

- **District Heating Temperature and Flow Velocity Control** *(highest priority, full validation pipeline):* Optimizes supply temperature and flow velocity using model predictive control to ensure sufficient heat delivery and minimize losses.

- **Demand-Side Flexibility Exploitation** *(low priority):* Uses dynamic pricing and AI-based consumer response prediction to shift heating demand to more cost-effective periods. This use case has a lower priority but is a candidate for future integration.

The first control use case is the core of the Danish node's TEF Test Use Case and will be demonstrated across the full validation chain. It serves as a benchmark for how AI can be reliably integrated into district heating operations. The second control use case, while not prioritized for immediate demonstration, offers promising potential for future testing, especially in scenarios involving consumer behavior and market-based control.

Operational Optimization Use Cases: These use cases aim to improve the overall efficiency and sustainability of energy system operations. They are currently considered with low priority. However, they are supported by the same infrastructure as the control and forecasting business use cases and may be integrated in the future.

- **Co-Optimization of PV Arrays and Electric Boilers** *(low priority)*: Uses PV forecasts to operate electric boilers when excess solar energy is available, reducing biomass consumption.
- **Optimization of District Heating Plant Operations** *(low priority):* Continuously adjusts plant setpoints using data-driven methods, potentially leveraging Physics-Informed Neural Networks (PINNs) for enhanced efficiency.

For both optimization business use cases, the full TEF validation pipeline will be considered for demonstration purposes. These use cases are particularly relevant for future scenarios involving sector coupling and integrated energy systems, and they offer valuable opportunities to showcase the node's capabilities in real-time optimization and sustainable energy planning.

Data flow

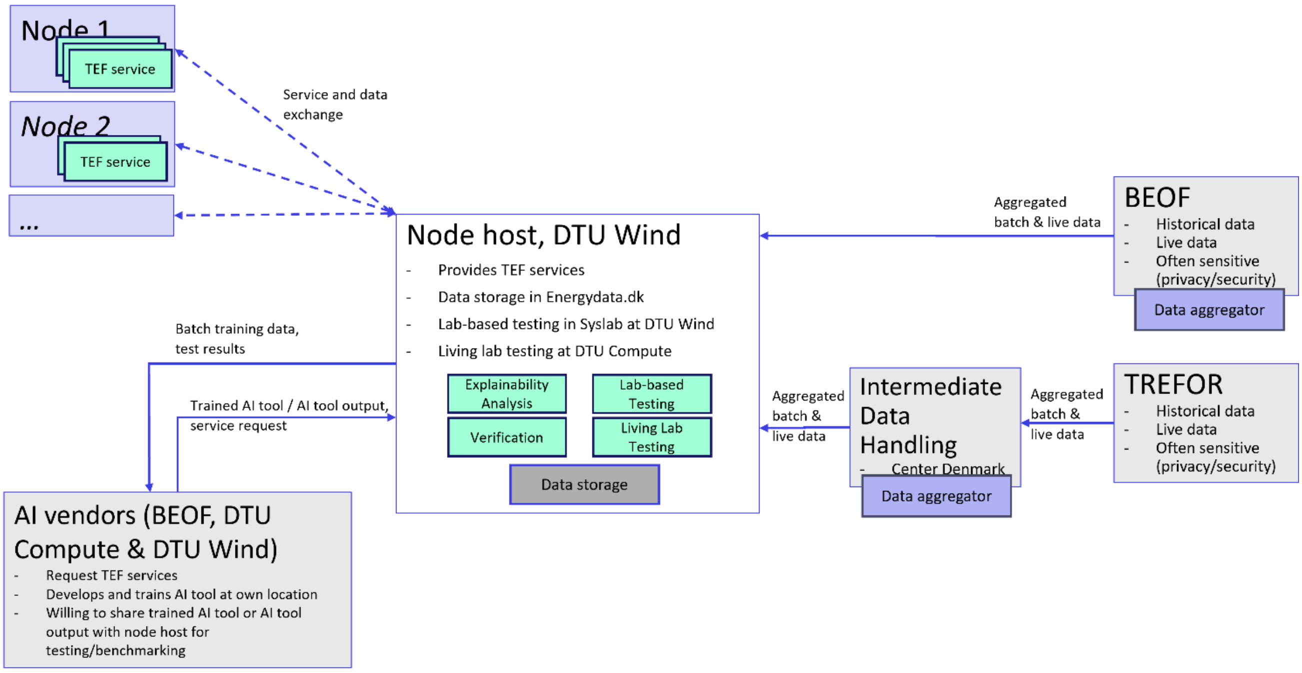


Figure 1: Overview of the Data Flow at and participants of the Danish node

As shown in Figure 1, the stakeholders of the Danish Node include the data providers (TREFOR Varme and BEOF), the intermediate data handling agency (Center Denmark) and the node host (DTU Wind), and the AI vendors (DTU Compute, DTU Wind and BEOF). Note that while DTU Wind hosts the node and uses their laboratory syslab for provision of lab-based testing and the energydata.dk platform for data handling and storage, DTU Compute hosts a living laboratory environment that will also be connected to the node. The data flow is designed to support lab-based and living-lab testing environments for the TEF Test Use Case.

The data flow begins with the data providers, BEOF and TREFOR Varme, who provide real-time consumer heat demand data. Although TREFOR is not a direct partner in the AI-EFFECT consortium, access to its data is enabled through Center Denmark, which acts as the data handling platform for TREFOR data. Center Denmark aggregates and enriches the data with weather forecasts (air temperature, wind speed and direction, solar radiation) and provides secure access to DTU via APIs. This enriched dataset is used for training and validating AI models. BEOF contributes historical smart meter data, which is, similar to TREFOR data, aggregated before sharing to ensure privacy compliance. BEOF also develops AI tools using its own data and shares both the models and relevant datasets with DTU Wind for service demonstration.

To ensure secure and efficient data exchange, the Danish node relies on several key enablers:

- Center Denmark's API and Delta Sharing Protocols facilitate structured and secure access to real-time and historical data.
- FMUs (Functional Mock-up Units) may be used to encapsulate AI tools for standardized integration into TEF services.
- Data aggregation is applied to the extent that it provides sufficient privacy protection while maintaining data utility for AI development.

The rough workflow including the TEF Services is described in the following:

- Real-time and historical data is collected and aggregated by Center Denmark and EnergyData.dk.
- AI vendors (DTU Compute, BEOF and DTU Wind) use the data to train forecasting and control algorithms.
- The trained AI tools are subjected to service provision incl. explainability analysis, verification and lab-based testing at DTU Wind.
- Lab-based testing is conducted at SYSLAB, a DTU Wind facility, where some of the AI tools are deployed in a controlled physical environment simulating district heating and power systems.
- Living-lab testing is performed at DTU Compute using real consumer data in an open-loop setup, comparing AI-generated setpoints with actual system behavior.
- AI vendors receive feedback and performance metrics, refine their models, and may re-submit improved versions for further testing.

This structured data flow enables robust, secure, and scalable validation of AI tools, supporting the Danish node's role in advancing intelligent control of multi-energy systems.

Challenges and mitigation

The development and operation of the Danish node within the AI-EFFECT TEF platform presents several challenges, both in building the node infrastructure and in running the TEF services effectively. These challenges span data access, service integration, technical alignment, and coordination across stakeholders.

One of the key challenges during node development is data access and privacy compliance. While the Danish node benefits from access to real-world district heating data from TREFOR Varme, TREFOR is not a direct partner in the AI-EFFECT consortium. Data access is enabled through Center Denmark, which requires careful coordination and the establishment of data-sharing agreements. Ensuring GDPR compliance while maintaining the utility of data for AI development is a continuous concern. To mitigate this, the node relies on data aggregation rather than synthetization, ensuring that individual consumer identities are protected while still enabling meaningful analysis.

Another challenge is the alignment between mathematical models and physical infrastructure. AI tools are often developed and verified using digital models, but discrepancies can arise when these tools are deployed in real-world lab environments. For example, measurement noise, communication delays, or unmodeled dynamics in SYSLAB may lead to deviations from expected behavior. This is mitigated through continuous testing and co-development of models and services, ensuring that digital representations are refined based on empirical observations.

Additionally, there might be a potential misalignment between verification results and real-world performance. AI tools that pass mathematical verification may still behave unexpectedly under real-world conditions. To mitigate this, the node ensures that verification is followed by rigorous lab and living-lab testing, and that any discrepancies are documented and used to refine both the models and the testing procedures.

The integration of the selected TEF services into a coherent and modular pipeline also presents logistical and technical challenges. Each service has different requirements for data formats, tool interfaces, and security protocols. To address this, the Danish node uses FMUs to standardize AI tool packaging and data sharing protocols to streamline data exchange.

During TEF operation, additional challenges may arise. One is the risk of delayed delivery of AI tools from vendors, which can impact the timeline for testing and demonstration. This is mitigated through early communication, fallback to simpler algorithms when needed, and flexible scheduling of service demonstrations.

Finally, cross-node collaborations such as reusing TEF services or sharing data between nodes—can be complicated by differences in data formats, privacy requirements, and infrastructure. The Danish node addresses this by ensuring that only aggregated data is retained locally and by developing structured APIs and interfaces that support interoperability.

Through these mitigation strategies, the Danish node ensures that challenges are addressed proactively, enabling robust, secure, and scalable validation of AI tools for multi-energy systems.

Future scope

The TEF Test Use Case at the Danish node is designed with modularity and scalability in mind, allowing for future enhancements and extensions beyond the current scope of the AI-EFFECT project. While the current setup focuses on validating AI tools for forecasting and control, several realistic ideas have been identified for expanding the node's capabilities.

One potential direction is the integration of additional AI tools targeting other business use cases already identified by the node, such as co-optimization of PV arrays and electric boilers, or demand-side flexibility exploitation. These tools could gradually be incorporated into the TEF validation pipeline, leveraging the existing infrastructure for explainability, verification, and testing.

On the infrastructure side, the node may be enhanced with additional physical components at SYSLAB, such as more advanced heat exchangers, variable-speed pumps, or hybrid energy sources. These additions would allow for more complex testing scenarios and better representation of real-world district heating systems.

There is also interest in developing digital twins of the physical infrastructure, enabling more flexible simulation-based testing and facilitating cross-node collaboration. This would allow AI tools to be tested virtually before being deployed in the lab, improving efficiency and reducing setup time.

Furthermore, the node may explore real-time integration with energy markets, enabling testing of AI tools that respond to dynamic pricing signals or grid conditions. This would be particularly relevant for future use cases involving sector coupling and demand-side flexibility.

While these ideas are not commitments, they represent realistic and strategically aligned opportunities for the Danish node to evolve and continue supporting innovation in AI-driven energy systems.

2.4 Dutch Node

The Dutch node is hosted by TU Delft. The main idea of the node is to bring together the transmission system operators (TSOs) and AI vendors and enable data sharing between them. This should enable the testing and development of AI tools for congestion management in transmission systems, which is the focus of the single TEF Test Use Case of this node. The node features mainly computational and software assets, as well as a testing facility for human-AI-interaction. It should be noted that due to ongoing node and platform planning, the TEF Service human-AI-interaction testing is not fully specified yet

The narrative of the TEF Test Use Case: Congestion management within transmission systems

The central ambition of this facility is to hone AI-based algorithms for congestion management, a critical function of control rooms. By providing a digital-physical and controlled environment replete with processes for synthetic grid data generation, the facility would allow for rigorous testing and refinement of these algorithms, ensuring they are robust and effective in managing the grid's congestion, e.g., through sequential topological reconfigurations.

The node includes the TEF Services data synthetization, benchmarking, human-AI-interaction testing and the corresponding experimental design. Details about the human-AI-interaction testing and the other TEF Services can be found in the Table 3 in the Appendix.

Corresponding Business Use Case(s)

The corresponding Business Use Case is called *"Support TSOs in the operation of the transmission system with AI-powered algorithms for decision-making in congestion management".* The TEF Test Use Case is built around the Business Use.

An AI tool addressing this use case should provide decision support for the human operator of the TSO in form of redispatch actions or switching actions that change the topology of the transmission systems.

Data flow

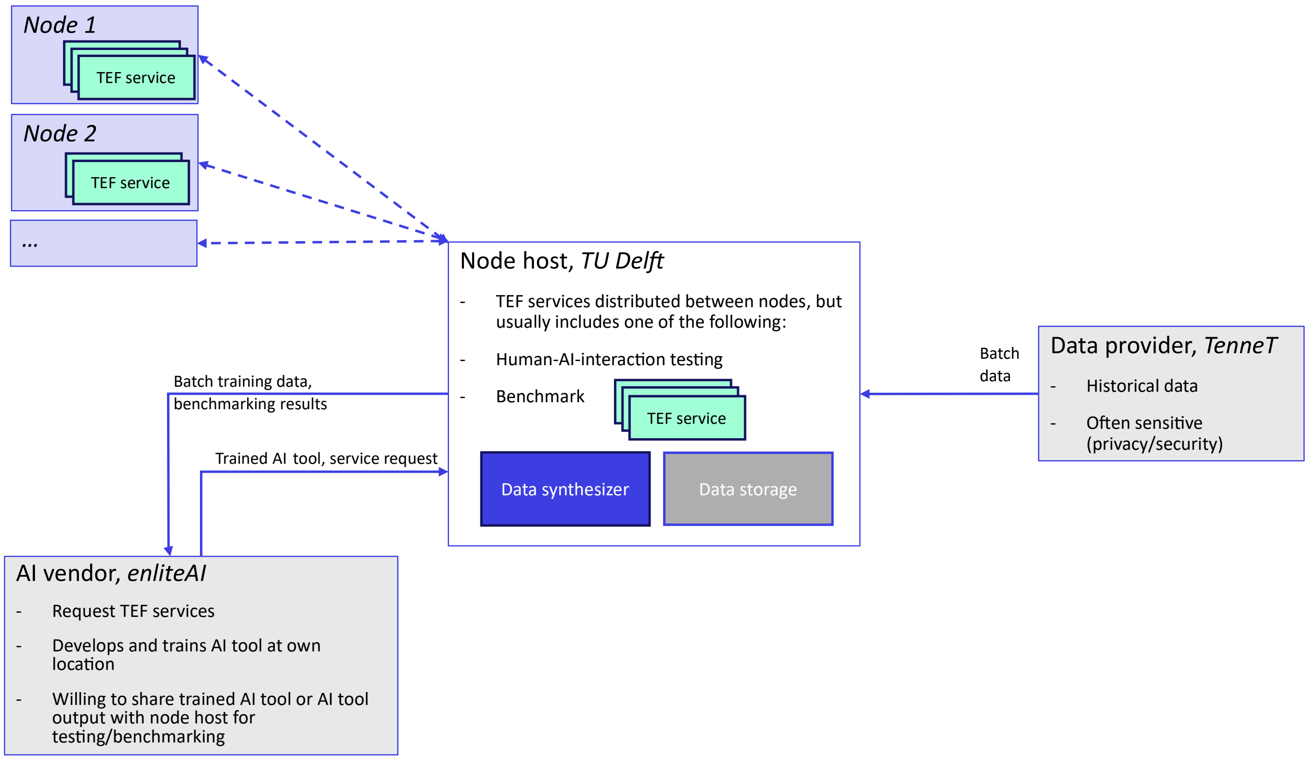


Figure 2: Overview over the data flow at the Dutch node

As shown in Figure 2**Error! Reference source not found.** the stakeholders of the Dutch node are the data provider (TSO), the node host and the AI vendor. As the main challenge for this TEF Test Use Case is the data sharing between the TSO and the AI vendor, the data synthesizer is a key element enabling the AI vendor to train its AI tool on synthetized TSO data. For developing and operating the data synthesizer a Non-Disclosure-Agreement (NDA) is set up between the node host and the TSO. The synthetized data can be handled as public data, thus no NDA is needed between the AI vendor and the other stakeholders. All data is shared in batches; no real-time data sharing protocols are required. Data exchange between the stakeholders may be established using APIs, the TSO data may be transferred to the node host as a one-time batch data transfer.

The rough workflow including the TEF Services is described in the following:

- TSO data is synthetized by the data synthesizer, creating a dataset of diverse transmission grid congestion cases defined by grid connectivity, substation design, circuit parameters and injections.
- AI vendors can use the data to train and test its AI algorithm. For this, the data is downloaded from the node host/AI-EFFECT platform.
- The trained AI tool for congestion management is deployed in the TEF and benchmarked against a chosen reference algorithm or a ranking list.
- The environment of real control room is imitated by a simulator for the transmission grid and a user interface for human interaction.
- AI-human interaction through a user interface is tested at the TEF, experiments for this are designed accordingly.
- The user can refine the algorithms and user interfaces and benchmark/test again.
- (User goes back to real control room and deploys the algorithms there, improves the usability and runs them).

Challenges and mitigation

Most of the challenges of this node are related to data sharing between the stakeholders. This is due to the sensitive data of the transmission system that is needed to train the AI tool on. To mitigate this, the data is synthetized before it is shared. In this way, the original data cannot be reverse-engineered from the synthetic data.

Another challenge is the data sharing of the AI tool of the AI vendor with the node host for benchmarking. Benchmarking needs to take place in a controlled environment including hardware, which is set up by the node host. The AI vendor therefore needs to share the tool with the node host. To circumvent sharing the whole source code of the tool, the vendor could share only the trained AI tool. The setup could follow a challenge-based approach, similar to the L2RPN or Flatland challenges.

Access to the node from external parties besides the node host could pose a problem. If the node is hosted at TU Delft in form of a virtual machine (VM), parties from outside TU Delft cannot directly interface the VM. A possible solution could be to set up a cloud with an interface to the VM.

The benchmark and the AI tool could aim at different objectives or base on different problem formulations. The circumvent this, a common problem formulation will be developed and shared with the AI vendor beforehand.

Concerning human-AI-interaction testing, an expert operator of a control room of a TSO would ideally test the human-AI-interaction of the developed AI tool. As this may not be possible, a testing methodology will be developed.

Future scope

In the future, the testing and validation capabilities of the node can be extended. Therefore, SCADA and RTDS may be connected to the simulation environment to make it more realistic and to also test the human-machine interaction as it would be in a real control room.

Concerning scalability, the benchmark model and computational resources should be scalable to various grid topologies and sizes. Also, the benchmark should be adaptable (open-source) and it should be possible to integrate other benchmark models.

2.5 German Node

The German node is hosted in Germany by Fraunhofer and operates as a distributed setup connecting a data provider, an AI vendor, and the node host. The use case targets congestion management in distribution networks with high shares of distributed renewable energy and increasing loads. The node offers primarily computational and software capabilities to provide a simulated distribution-system environment for AI testing. This supports the AI-EFFECT goal of a trusted, interoperable TEF platform for validating AI solutions that enhance DSO decision-making and secure DER integration.

The narrative of the TEF Test Use Case: Distribution Network Congestion Management for Renewable Integration

The TEF provides a simulated distribution-network environment fed by anonymized operational data to evaluate AI for congestion management with high DER and increasing loads. AI agents analyze grid states, predict congestion, and propose actions to mitigate issues while respecting operational constraints and, where relevant, incorporating bird-activity considerations.

TEF Services are integrated into a single workflow: Data Provision enables secure data onboarding and exchange; AI-Tool Benchmarking executes controlled, repeatable experiments; Explainability Analysis offers human-interpretable insights into AI decisions; and Mathematical Verification validates constraint compliance and robustness across scenarios. The resulting evidence supports DSO decision-making and readiness assessments for deployment.

Corresponding Business Use Case(s)

Enable DSOs to maintain reliable, high-quality service on distribution networks with growing DERs and loads by proactively managing congestion. AI supports operational decisions to anticipate and mitigate overloads within operational and environmental constraints, allowing greater renewable integration, and reduced outages.

Data flow

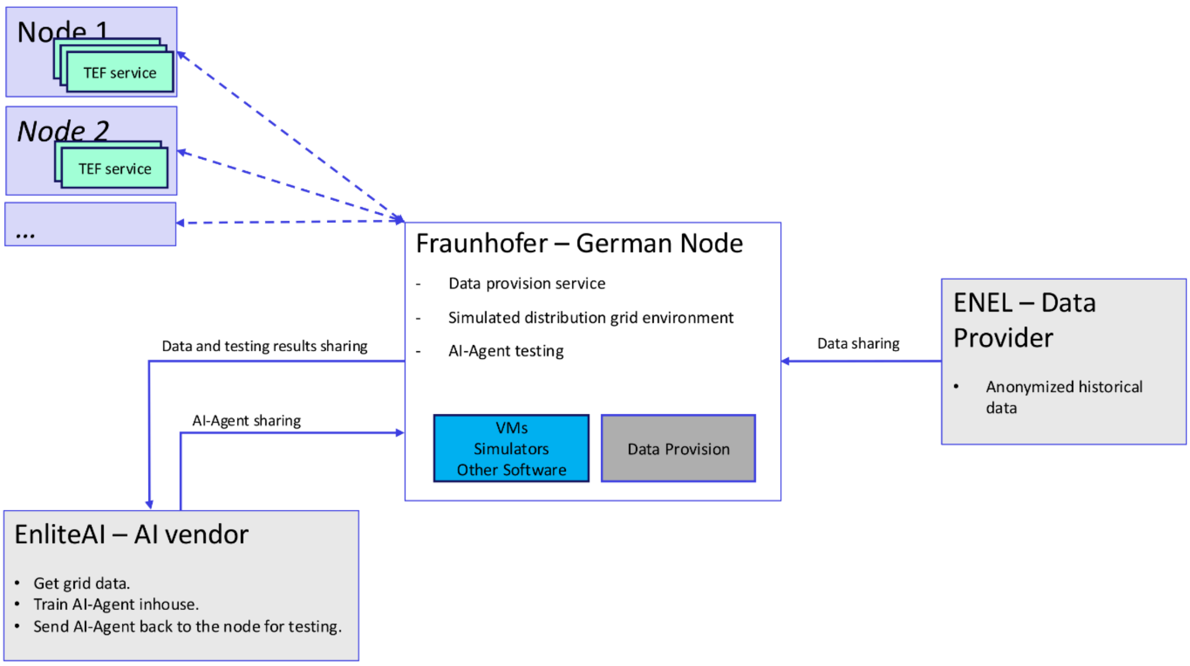


Figure 3: Overview over the data flow at the German node

Figure 3 shows the workflow as following:

- Fraunhofer node hosts the virtual environment as well as the needed simulators
- ENEL provides data (historical datasets) for testing, validation, and simulation
- EnliteAI is the AI vendor that will train the agent in-house and provide the trained agent back to the node for testing and validation of the performance
- The node may communicate with other nodes for shared utilities or requesting services provided by other Nodes.

Challenges and mitigation

For the German node, sharing data outside the node is not intended, any external transfer would only be considered after explicit approval from the data provider and after applying data anonymization, or encryption, along with clear access controls.

A technical challenge arises from a solver mismatch: Grid2Op, the framework used to train the AI agent, natively expects pandapower, whereas the node operates with the OpenDSS simulator from EPRI, to bridge this gap, the team will implement a custom backend adapter that maps Grid2Op's required interfaces, observations, and actions to OpenDSS models and results, ensuring functional parity and consistent performance.

Additionally, if an AI vendor must train models on-premises without sharing source code, the node will adopt a challenge-based interface where the vendor trains locally and delivers a packaged agent the node will then validate and benchmark the agent within its environment under controlled scenarios.

2.6 Portuguese Node

The Portuguese node is located primarily at INESC TEC's X-Energy Lab and supported by CEVE and Watt-IS. The node focuses on enhancing energy efficiency, management, and sharing within local energy communities. The node supports use cases that join AI vendors with DSOs, and integrates both physical (EV chargers, battery storage, controllable loads) and virtual assets. The node features simulators, digital twins, and hardware-in-the-loop setups, providing an environment for developing and testing AI solutions in the energy sector. Moreover, the node prioritizes the deployment and use of a data space and necessary connectors for improved data sovereignty, where consumer data (among other types of data) will be accessible. This setup aligns with the AI-EFFECT project's goal of creating a practical Testing and Experimentation Facility platform.

2.6.1 Performance evaluation of AI-powered Virtual Energy Manager

The narrative of the TEF Test Use Case

This TEF Test Use Case explores the integration of AI-powered tools to promote energy efficiency at the residential level. The central component is WIS4Households, an intelligent solution that utilizes smart meter data provided by Distribution System Operators (DSOs) to deliver targeted and quantified energy efficiency recommendations to end-users. The tool operates through the implementation of AI-based Non-Intrusive Load Monitoring (NILM) algorithms, which estimate the energy consumption of individual household appliances without the need for additional hardware installation.

The TEF Use Case includes the TEF Services Data Provision and benchmarking.

Corresponding Business Use Case(s)

The corresponding Business Use Case is called "AI-powered Virtual Energy Manager tool (WIS4Households) to support residential users in improving energy efficiency". The TEF Test Use Case is built around this Business Use Case and demonstrates how smart meter data, combined with AI-based algorithms such as Non-Intrusive Load Monitoring (NILM), can be used to generate targeted and quantified energy efficiency recommendations. The TEF Services "Data Provision" and "AI Tool Benchmarking" support the implementation and validation of this Business Use Case.

Data flow

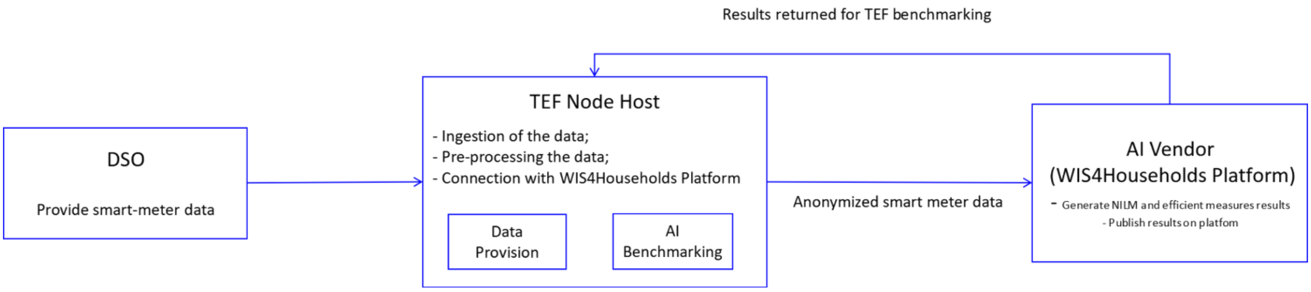


Figure 4: Overview of the data flow of the first TEF Test Use Case at the Portuguese node

As shown in the Figure 4, the TEF Test Use Case centers on the operation of the WIS4Households platform within a TEF node. The data flow involves several key stakeholders:

- **Electricity Retailer / DSO:** Provides 15-minute interval smart meter data in anonymized form.
- **TEF Node Host:** Hosts the TEF node, supports data handling, and contributes to AI development.
- **AI Vendor:** Develops the web interface and integrates the AI models (NILM and recommendation modules).

Data Flow Overview:

- Smart meter data is securely ingested into the TEF node.
- Data is preprocessed (timestamp alignment, format normalization, anonymization).
- The NILM algorithm disaggregates appliance-level consumption using supervised learning (regression).
- A second module generates energy efficiency recommendations.
- Users view results via a web interface.
- AI Tool Benchmarking evaluates model accuracy and robustness.

Key enablers include APIs, local deployment, containerized modules, and GDPR-compliant data handling.

Challenges and mitigation

Key challenges in this TEF Test Use Case include delivering meaningful insights to residential users and ensuring large-scale adoption. WIS4Households addresses this by providing appliance-level consumption data and targeted energy efficiency recommendations, going beyond typical HEMS capabilities.

Scalability is tackled by using only existing smart meter data, avoiding the need for additional hardware. Data privacy is ensured through anonymization and local processing within the TEF node. To guarantee reliability, the AI models are tested for robustness against missing or noisy data.

Future scope

In the future, WIS4Households may be enhanced with forecasting tools for energy consumption, production, weather, and market prices. This would allow users with variable tariffs to receive optimized appliance usage recommendations for upcoming days.

Given that the system relies only on existing smart metering data and requires no additional hardware, it is highly scalable. The solution can be deployed across any energy grid equipped with an Advanced Metering Infrastructure (AMI), making it suitable for large-scale rollout.

2.6.2 Performance evaluation of energy-sharing mechanism

The narrative of the TEF Test Use Case

This TEF Test Use Case evaluates the performance of AI-based energy-sharing optimization within an energy community. It aims to maximize economic benefits and fairness by optimizing energy-sharing coefficients based on consumption, generation, and surplus forecasts.

The TEF Services integrated include Data Provision and AI Tool Benchmarking. These services enable fair and efficient energy distribution, increasing participant savings and reducing energy surplus.

The use case validates the impact of AI-driven optimization on economic, operational, and environmental metrics, benefiting community participants and energy suppliers through improved energy management and cost reduction.

Corresponding Business Use Case(s)

The corresponding Business Use Case is "AI-based optimization of energy-sharing in energy communities." The TEF Test Use Case demonstrates how forecasted smart meter data such as consumption, production and surplus, previously trained using historical data, can optimize sharing coefficients. This optimization will increase the shared energy between community participants and maximize financial savings.

TEF Services "Data Provision" and "AI Tool Benchmarking" support this by enabling data access and validating model performance through KPIs such as increased participant savings and shared energy volume.

Data flow

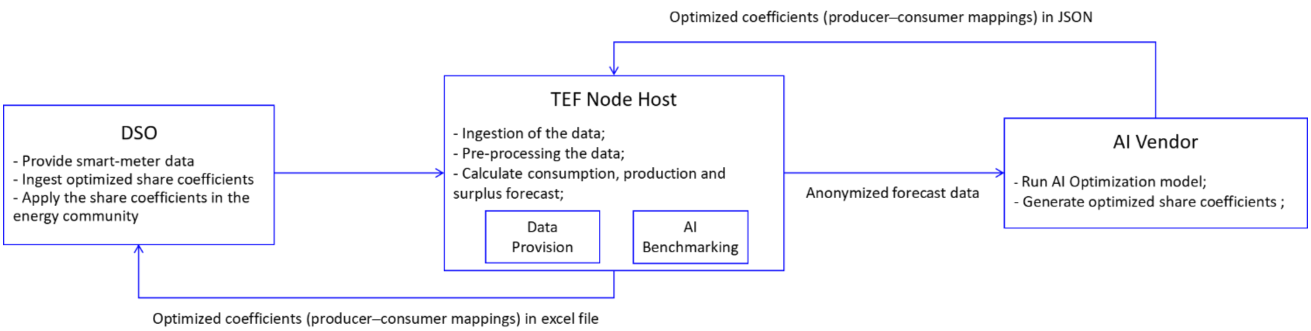


Figure 5: Overview of the data flow of the second TEF Test Use Case at the Portuguese node

The TEF Test Use Case centers on the local execution of an AI-based energy-sharing optimization model within a TEF node as shown in Figure 5. The data flow involves several key stakeholders:

- **Electricity Retailer / DSO:** Provides historic, 15-minute interval smart meter data, including consumption and surplus per participant. Due to security constraints, direct data exchange with other actors is restricted. Also responsible for receiving and applying the optimized sharing coefficients in the energy community.
- **Node Host:** Hosts the TEF node, manages secure data handling, performs data pre-processing, and supports AI development, process forecasts based on historical data.
- **AI Vendor:** Develops and runs the AI optimization model locally within the node.

Data Flow Overview:

- Historic smart meter data is securely ingested into the TEF node.
- Data is preprocessed (e.g., timestamp alignment, JSON formatting).
- The AI model, based on linear programming, optimizes energy-sharing coefficients to maximize savings and minimize surplus.
- The model receives input in JSON format and returns optimized coefficients (producer–consumer mappings) also in JSON.
- Optimized coefficients are shared with the DSO to be tested in a simulated real-time environment reflecting energy community operations, including flexible assets.

Key enablers include RESTful APIs, local deployment, containerized or VM-based execution, data anonymization, and alignment with GDPR and security protocols.

Challenges and mitigation

Key challenges in this TEF Test Use Case include ensuring fair energy distribution among community participants and enabling scalability across different community sizes and infrastructures. The AI model addresses fairness by applying transparent allocation rules based on consumption, generation capacity, and available surplus, ensuring equitable outcomes for all users.

Scalability is ensured by designing the optimization model to operate with standard smart meter data, without requiring additional hardware. Interoperability is achieved through the use of standardized data formats and RESTful APIs.

Future scope

The TEF Test Use Case may evolve toward real-time optimization of energy-sharing coefficients, reducing reliance on forecast data. Future integration of additional TEF Services—such as Real-Time Monitoring or Federated Learning—could enhance performance and adaptability.

The optimization model is designed to scale and can be extended to larger communities, including urban districts or regional grids. With proper system integration, the AI-based approach can support thousands of participants, enabling more efficient and flexible energy distribution on a broader scale.

2.6.3 Testing of DER scheduling/control algorithms

The narrative of the TEF Test Use Case

This TEF Test Use Case is designed to provide a structured, multi-layered environment for testing and validating AI-based control and scheduling algorithms for DERs operating in low-voltage (LV) distribution networks. It addresses both technical challenges (e.g., voltage violations, underutilization of renewable generation, inefficient flexibility activation) and operational constraints (e.g., limited observability, data privacy, and user comfort requirements).

The core objective is to enable TEF users, including AI developers, DSO innovation teams, and technology providers, to enhance, benchmark, and stress-test their control and scheduling algorithms under conditions that closely mimic the complexities of real-world LV grids. The environment supports algorithms with multiple potential objectives:

- Voltage control – Minimize active and reactive power adjustments required to prevent or correct voltage limit violations, ensuring compliance with grid codes.
- Dynamic Operating Envelopes (DOEs) – Define and update safe, time-varying flexibility bounds for each controllable resource, aligned with network constraints and forecasts.
- Renewable utilization maximization – Optimize DER operation (e.g., EV charging schedules) to absorb maximum available renewable generation without impacting user comfort or creating new technical violations.
- Distributed/edge intelligence – Implement control strategies directly on DER interfaces (e.g., EV chargers, battery management systems) to reduce communication overhead and protect private data.

Once developed, these models are connected to the testing environments, whether virtual, physical, or a mix of both, hosted at the node (INESC TEC), accessed via REST APIs, featuring:

- Data Provision and Synthetization – Operational datasets (e.g., load, PV generation, voltage measurements, flexibility requests) or privacy-preserving synthetic equivalents.
- Performance Evaluation and Benchmarking – Algorithms are compared against both baseline models (simple rule-based or optimization heuristics) and state-of-the-art references from academic and industrial contexts.
- Mathematical Verification – Algorithms are stress-tested against the optimal control solution.
- Simulation-Based Testing – Algorithms are evaluated across a wide range of operating conditions.

- Lab-Based Testing – The TEF provides access to a controlled physical environment where real DER assets (e.g., batteries, EV chargers, smart appliances) and digital twins, interact with the AI tool, enabling validation under true hardware dynamics, latencies, and device-specific constraints.

Corresponding Business Use Case(s)

The Test Use Case covers two Business Use Cases:

1. **Voltage control with distributed energy resources** – AI-based control strategies for DERs that detect, prevent, or mitigate voltage violations in LV grids, either in real-time or through preventive scheduling.
2. **Edge load control for flexibility services** – Distributed, edge-based control solutions for flexible loads (e.g., storage, electric water heaters, heat pumps, EV chargers), enabling the provision of grid services and participation in flexibility markets while respecting user-defined comfort constraints.

Data flow

The interaction between the main stakeholders and technical components involved in the Portuguese node for the DER scheduling/control Test Use Case are as follows

- Data Providers (e.g., CEVE) supply historical and near-real-time operational datasets, including load and generation measurements, voltage profiles, renewable production forecasts, flexibility event requests, and relevant tariff or market data. This data originates primarily from smart meters, secondary substations, and other field monitoring equipment. Parts of the grid will also be used to build digital twins for the lab-based testing.
- Node Host (INESC TEC) acts as the TEF service orchestrator, ensuring secure and structured data exchange between providers and AI vendors. It manages data anonymization, controls access rights, performs synthetic data generation when needed, and maintains both the virtual testing facilities (digital twins of LV networks) and the physical lab environments where real DER assets can be interfaced.
- AI Vendors (e.g., INESC TEC research teams, Watt-IS, or other external developers) design, train, and refine DER control and scheduling algorithms. These algorithms are then deployed into the TEF's virtual or lab-based environments via standardized APIs, allowing their performance to be tested under diverse simulated and real-world operating conditions. Vendors can also directly interact with the TEF to request benchmarking, mathematical verification, or additional data for model improvement.

The data flow is bidirectional:

1. From TEF to AI Vendors – Provision of operational or synthetic datasets, simulation environment access, and performance feedback from previous testing rounds.
2. From AI Vendors to TEF – Connection of trained models, control setpoints, or simulation/lab-based test results for benchmarking and verification.

This closed-loop process enables iterative refinement: control algorithms are continuously improved based on benchmarking outcomes, constraint verification reports, and feedback from both simulation and hardware-in-the-loop experiments. The data flow can be represented graphically in a very similar way to the one considered in Figure 6.

Challenges and mitigation

The main challenges and their mitigations include the following: Incomplete or poor-quality operational data can be addressed by implementing strict database validation, incorporating redundancy through alternative data sources, and using robust imputation and outlier detection mechanisms. To ensure synthetic data realism, quantitative quality metrics should be applied to compare synthetic datasets with operational equivalents, focusing on statistical and time-series similarity. When selecting relevant benchmarks, it is best to start with transparent, simple baselines such as linear regression or rule-based control, and then incrementally include advanced, state-of-the-art models from the literature. For the integration of AI control tools with physical assets, containerized deployments and standardized communication protocols can help reduce setup complexity. Finally, guaranteeing real-time execution requires enforcing runtime constraints during testing to ensure that execution times remain below data resolution intervals.

Future scope

The future development of this TEF Test Use Case foresees the integration of Conformity Assessment services to verify compliance of DER control algorithms with industry standards, cybersecurity requirements, and the EU AI Act. Expanding the test environments to additional LV and MV grids (e.g., using equivalent models) would increase diversity in operational scenarios, enabling the evaluation of scalability and adaptability. The inclusion of edge computing is envisioned to test deployment conditions where processing occurs directly at DER devices or local controllers. The living lab dimension could be extended to energy communities and prosumer networks, enabling the validation of control strategies under real multi-actor, market-driven conditions. On the simulation side, extreme-event and fault-condition stress testing would provide insight into the robustness of algorithms against rare but critical disturbances.

2.6.4 Validation of Load and RES forecasting algorithms

The narrative of the TEF Test Use Case

The TEF Test Use Case focuses on validating and improving the performance of AI-based Load and Renewable Energy Source forecasting algorithms for energy management. The primary objective is to create an environment where TEF users and AI developers can enhance and rigorously test their forecasting models. This involves providing access to operational and external data, relevant features, and performance assessment tools. The narrative involves deploying a load or RES forecasting algorithm in the TEF, requesting domain-specific or synthetic data, accessing additional features from a Knowledge Store, and evaluating the model's performance. The user can then refine the model iteratively with additional data and performance evaluation information. More specifically, the following TEF services are expected to be integrated: Data Provision, Data Synthetization, Knowledge Store, AI Tool Benchmarking, Feature Importance Analysis, Mathematical Verification. Additionally, we aim to use a TEF service that relates to the Energy Consumption of AI Algorithms, although not a priority, if deployed during the development phase of the project.

Corresponding Business Use Case(s)

This TEF Test Use Case related to two Business Use Cases. The first one is named “Load and RES Generation forecasting”, which covers the development and validation of AI models for load and RES generation forecasting, ensuring they are robust and perform well under various conditions. These forecasting algorithms support energy management and decision-making, for instance in households and energy communities.

The second one is named “Forecasting Electric Vehicle Charging Patterns”, which focuses on the specific application of load forecasting for electric vehicle charging patterns, which is a subset of the broader load forecasting use case. These forecasting algorithms support energy management and decision-making in electric vehicle charging stations.

Data flow

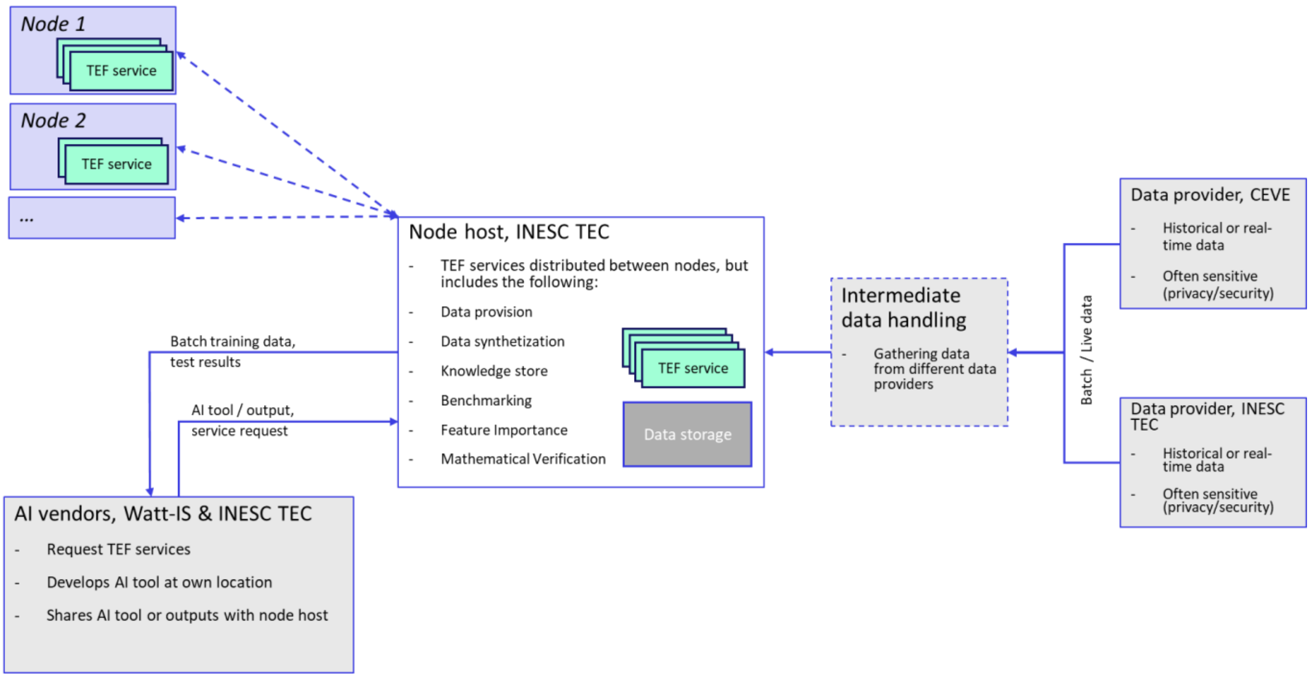


Figure 6: Overview of the data flow of the fourth TEF Test Use Case at the Portuguese node

Figure 6 ***Error! Reference source not found.***illustrates the interaction among various stakeholders and components within the Portuguese node, which is connected to the AI-EFFECT central platform. The key stakeholders include AI vendors, more specifically Watt-IS and INESC TEC, which can request the execution of TEF services, and are responsible for developing and training AI tools at their own locations and sharing these tools or their outputs with the node host. The node host, managed by INESC TEC, provides TEF services, manages and provides access to physical and virtual assets, and stores data, acting as an intermediary between AI vendors and data providers to facilitate the exchange of data and services. Data providers, including CEVE and INESC TEC, supply historical or real-time data necessary for the development and testing of AI tools. The data exchanged in this Test Use Case is mainly related to load and generation of residential consumers. The exchange of data, models, and outputs are made possible via APIs available within the node.

Challenges and mitigation

The development of the Test Use Case may encounter challenges that need to be carefully managed. One challenge is maintaining the quality of operational data used, as high-quality data is crucial for training models. To mitigate this, it is essential to implement correct database management practices and ensure the correct handling of various operational data sources. Additionally, having redundancy with alternative data sources can help maintain data availability. Another challenge is ensuring the quality of data synthetization to reflect real-world conditions. Synthetic data is vital for augmenting the dataset when real data is scarce. To address this, it is important to implement synthetic data quality metrics that can evaluate the similarity between synthetic and operational data, ensuring the synthetic data is representative of real-world conditions. Selecting the right baselines for performance comparison is also a challenge. Baselines are necessary for benchmarking the performance of AI models. To mitigate this, we can start with common and simple models and conduct a literature review to identify relevant state-of-the-art models. This ensures that the baselines are relevant and provide a meaningful benchmark for evaluating model performance.

Future scope

The future scope of the TEF Test Use Case includes several enhancements and scalability improvements. One key enhancement involves operationalizing forecasting tools with simulation-based tools to compute model performance under real-world and purpose-specific conditions, providing more accurate insights. Additionally, integrating TEF services like Conformity Assessment can ensure AI solutions meet industry standards and legal requirements.

Ensuring compatibility with all kinds of forecasting techniques would broaden the TEF's applicability, making it a versatile platform. For scalability, developing a stress-testing environment to simulate extreme weather conditions would help assess the robustness of forecasting models, namely within the context of climate change.

2.6.5 Performance evaluation of network monitoring algorithms

The narrative of the TEF Test Use Case

This TEF Test Use Case is centered around evaluating and improving AI-based network monitoring algorithms in low-voltage (LV) distribution grids, which are typically under-monitored, poorly modelled, and subject to high variability due to the proliferation of distributed energy resources (DERs), including electric vehicles (EVs), and demand-side strategies. These conditions make real-time observability and accurate state reconstruction inherently challenging, especially where real-time metering is partial or absent.

The narrative reflects a complete lifecycle of AI model validation, structured around the modular capabilities of the TEF. It begins with the provision of operational data (historical voltage and power measurements, irradiance, topological data) from Distribution System Operators (DSOs), particularly considering the project partner CEVE. These datasets may be incomplete, noisy, or privacy-restricted, which may justify the need for data synthetization services to generate privacy-compliant synthetic datasets for model training or robustness testing.

AI vendors, such as INESC TEC researchers, Watt-IS, or third-party developers, will use these datasets to train and refine algorithms targeting:

- State estimation – reconstructing the real-time or near-future state (voltage, power injections) of a LV grid mostly unobservable in real time;
- Topology reconstruction – inferring or correcting the physical structure (connectivity and impedances) of the grid;
- Phase assignment – determining to which phase a given single-phase consumer is connected.

Once developed, these models are connected to the testing environments, whether virtual, physical, or a mix of both, hosted at the node (INESC TEC), accessed via REST APIs, featuring:

- Data Provision (historical and real-time smart meter data from DSOs, public sources or generated by the Lab Testing Environment);
- Data Synthetization;
- AI Tool Benchmarking (comparison with state-of-the-art or simple baselines);
- Virtual Facility Access (access to the Lab Testing Environment to enable interactions with physical components, along with digital twins and Hardware-in-the-Loop);
- Simulation-based Testing (simulations with multiple pre-defined scenarios, including custom setups).

Corresponding Business Use Case(s)

This Test Use Case corresponds directly to the following two Business Use Cases:

1. **State Estimation and Network Quality -** This involves the application of AI techniques to estimate voltages and power flows under conditions of partial observability, supporting grid operators in maintaining system reliability and quality metrics.

2. **Network Topology Estimation -** AI models are applied to infer missing or incorrect topology and cable characteristics, addressing the common issue of inaccurate or outdated grid configurations in LV systems. This includes phase identification and impedance correction.

Each of these business use cases aligns with specific components of the TEF Test Use Case: the first is covered by the development and evaluation of state estimation algorithms, and the second is supported by tools focused on grid topology and electrical characteristic discovery.

Data flow

The interaction between the primary stakeholders and components at the Portuguese node for the network monitoring Test Use Case are as follows:

- Data Providers (e.g., CEVE) supply historical and near-real-time data (voltage, active power, irradiance) from smart meters and substations.
- Node Host (INESC TEC) facilitates TEF services, including making data accessible to AI vendors via APIs, managing synthetic data generation, and orchestrating virtual environments for testing.
- AI Vendors (e.g., INESC TEC internal teams) develop and train algorithms, which are then deployed into the node's testing environment or used to request performance evaluations.

Data flows in both directions: vendors receive training data and push results or models back for benchmarking or simulation testing. The feedback loop enables refinement of the AI tools. The data flow can be represented graphically in a very similar way to the one considered in Figure 6.

Challenges and mitigation

The key challenges and corresponding mitigations include the following: Incomplete or low-quality operational data can be managed by enforcing database validation pipelines, ensuring redundancy with alternative data streams, and applying proper data imputation and outlier detection mechanisms. For the selection of valid benchmarking models, it is advisable to begin with simple, interpretable approaches such as linear regression and expand systematically through a structured literature review of state-of-the-art algorithms. Privacy concerns in data sharing can be mitigated by providing robust data anonymization and synthetization services, enabling realistic yet privacy-safe training. To address the integration complexity of AI tools with simulation, containerized environments with documented APIs and compatible platforms, such as Linux VMs and REST endpoints, can be offered to reduce overhead. Finally, the risk of real-time constraints not being met by AI algorithms can be mitigated by incorporating performance monitoring and stress testing during design, ensuring that runtime consistently remains below data resolution thresholds.

Future scope

The future scope of this TEF Test Use Case includes several enhancements aimed at increasing both the depth and breadth of its applicability in real-world LV grid scenarios. One important direction is the integration of a Knowledge Store service, which would enable the provision of additional exogenous features such as numerical weather predictions or socio-technical datasets, enhancing the predictive capability of state estimation and topology discovery models. Another promising development is the expansion of the testing framework to medium-voltage (MV) grids (e.g., via equivalent models), where more complex topologies and operational constraints would further stress-test the robustness of AI tools. From an operational perspective, embedding the testing and benchmarking process into edge computing platforms, namely in DERs, could allow continuous model validation in near-real-time conditions, with data privacy safeguards. The TEF could also incorporate Conformity Assessment services to align AI tools with the EU AI Act, sectoral technical standards, and cybersecurity requirements for greater trust and adoption readiness. These enhancements, while not commitments, represent realistic and strategically aligned evolutions that would strengthen the TEF's role as a reference platform for the validation, certification, and continuous improvement of AI-based network monitoring solutions.

3 Linking the TEF and the Platform

The project aims to create a digital platform that provides access to the TEF. This section summarizes important information of Section 2 to provide an overview and requirements for the interaction between the TEF and the platform. For this, the data flow at a generic node is analyzed in Section 3.1, the data sharing within the TEF is described in Section 3.2, and the interaction of the TEF and the platform are discussed in Section 3.3. Additionally, Section 3.4 introduces the approach to interoperability of AI-EFFECT.

3.1 Data Flow at a generic AI-EFFECT Node

Analyzing the documentation of the four nodes given in Section 2, three stakeholders can be identified for a typical AI-EFFECT node, which can be seen in Figure 7. As this reflects the current state of the node's development plans, future adjustments to the data flows displayed should be possible.

The **data provider** sends data to the node host to test and train the AI tool. We can distinguish between two types of data: live data (data provider A), which is measured and transferred in real time, and historical data (data provider B), which is transferred in batches. Both types of data are often sensitive due to privacy or security concerns related to personal energy usage or critical infrastructure, such as the transmission grid. For the Danish and the Portuguese node, the data delivered by the data provider undergoes an intermediate data handling step where data is gathered before being forwarded to the node host. Both nodes make use of data spaces here. As this is not the case for the other nodes, the intermediate data handling is framed in dotted lines in Figure 7.

The **node host** receives the data sent by the data provider and processes it, if not done by the data provider already, so that it can be published without raising privacy or security concerns. To accomplish this, the data can be synthesized, aggregated, or anonymized. Additionally, testing in the form of benchmarking, lab testing, or living lab testing takes place on the premises of the node host. The node host also provides other TEF services that are part of the TEF Test Use Case. A connection to the other nodes of the platform enables inter-nodal provision of TEF Services. It is important to note that a node host may consist of a group of stakeholders, each of whom provides specific infrastructure to the TEF.

The **AI vendor** develops and trains AI tools on its own premises and aims to test them. To test these tools, the vendor uses the TEF Services provided by the node host via a data connection. Some services may be run on the AI vendor's premises, in which case the necessary data is sent to the vendor. Others require the vendor to share the trained AI tool or at least the tool's output with the node host. Furthermore, the node host sends training data and test results to the AI vendor. Although the AI vendor is presented as the primary user of the TEF, other stakeholders should also be able to access and benefit from the TEF services. For instance, the data provider at the Dutch node may supply the original TSO grid data and use the TEF Service of data synthetization to receive the synthesized data. In the remainder of this text, we will use the term **TEF user** for this.

Besides data sharing between the stakeholders of the node, data can also be shared between the nodes, which is illustrated with dotted lines in Figure 7, and will be discussed in the next sections.

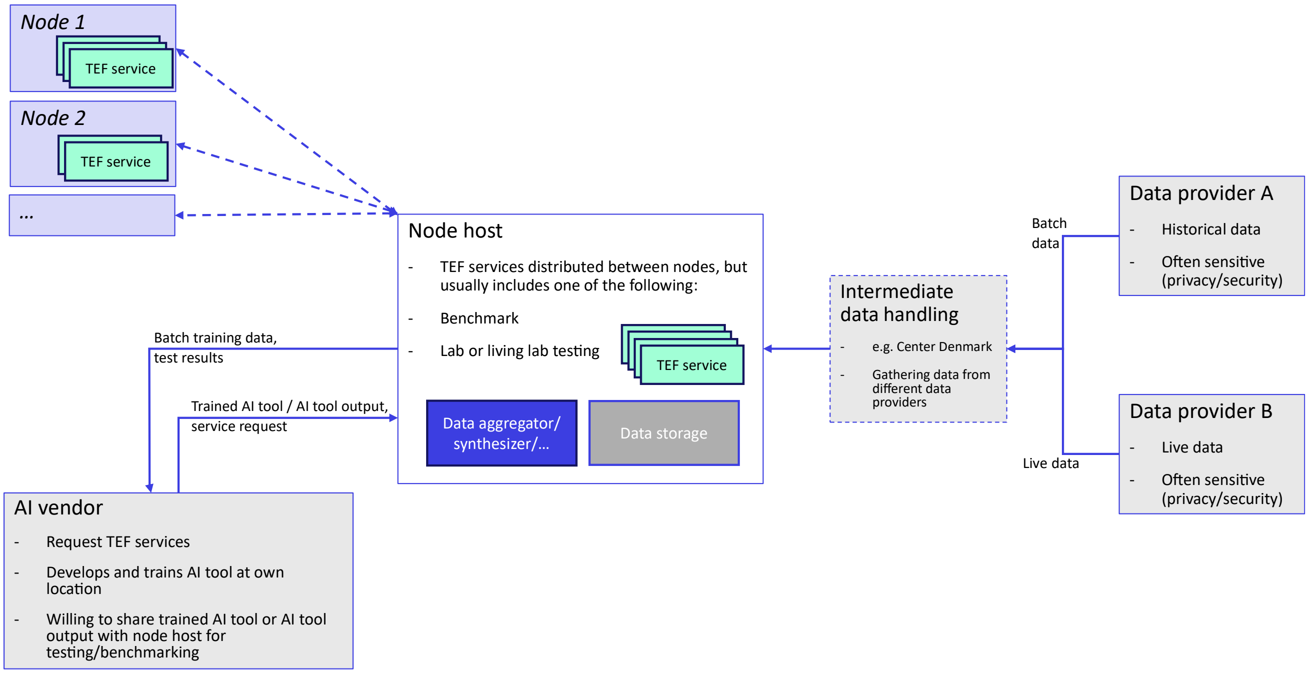


Figure 7: Overview of the data flow at a generic node of the TEF

3.2 Data Sharing

Data sharing within the TEF can be categorized into three hierarchical levels:

- **Internodal data sharing (highest level)** – Data sharing between different nodes of the TEF. An example is the explainability service of the Danish node, which can be accessed by other nodes.
- **Intranodal data sharing (intermediate level)** – Data sharing between stakeholders operating within a single node. An example is the data flow between the data provider and the node host.
- **Intra-stakeholder data sharing (lowest level)** – Data sharing between different components within a single stakeholder's environment, such as between individual laboratory or software assets. An example is the data flow between the different laboratory assets in the Portuguese node.

Data privacy and confidentiality are central concerns when sharing testing and training datasets. This applies to both personal data, which must comply with GDPR, and data from critical infrastructure, such as TSO datasets from the Dutch node. Hence, besides the technical realization of data sharing, the legal frameworks that enable data sharing, such as non-disclosure agreements (NDAs), need to be implemented. To enable secure and compliant data sharing while circumventing elaborate bilateral NDAs, methods such as data synthetization, aggregation, and anonymization can be applied to transform sensitive data into publicly shareable data. All data sharing protocols should be selected to suit the type of data and the sensitivity of the data.

For instance, the Danish node involves sensitive heat demand data from real district heating customers in the central part of Denmark that cannot be shared within the node, i.e., it is a requirement that no information about the customers can be inferred from the data shared within (or outside) of the node because of the General Data Protection Regulation (GDPR). In principle, it is possible to anonymize the data by removing information about addresses, locations, smart meter IDs, etc., which are used to connect the customer to their heat demand time series. However, some areas are sparsely populated and for a given use case, it might be possible to infer that a given heat demand time series corresponds to one of a few (e.g., five) customers. This is not acceptable, and even though anonymization may be a technically minimal mechanism (in terms of algorithmic complexity), it does not satisfy the requirement of anonymity. In contrast, a sufficiently coarse aggregation of customers' heat demand series may ensure anonymity, but depending on the aggregation approach (statistics-based, topology-based, location-based), it may not be minimal if it requires complex algorithms or human intervention.

Besides training and test data, the trained AI tools are confidential and must be handled with care. For some TEF Services, it may be sufficient to share only the outputs of the AI tools for testing purposes, so that the TEF queries the AI tools by sending inputs and receiving outputs from the TEF user. Here, co-simulation tools like the Functional Mock-up Interface (FMI) standard can be used. For other TEF Services, for example, benchmarking, it may be important to have the AI tool running on premises of the TEF. In such a case, only the weights of the trained AI tool can be shared, or the whole AI tool is containerized and encrypted, so that no information is leaked.

Most data exchanged within the platform is transmitted in batches, which does not require real-time data sharing protocols. However, some TEF Services require low-latency or close-to-real-time capabilities. One example is the living lab testing at the Portuguese node, where rapid data sharing between the TEF and the TEF user may be essential for the purpose of effective control of distributed energy resources.

Finally, to ensure a secure, standardized, and efficient data exchange between stakeholders, the Danish and the Portuguese node deploy data spaces. This not only ensures compliance with privacy and security requirements but also increases interoperability across the TEF. Section 3.4 discusses further details on interoperability measures.

3.3 The AI-EFFECT Platform

The TEF, consisting of the four nodes, will be complemented by a digital platform. The exact role and the specifications of the platform are still subject of discussion between the work packages WP1, WP2 and WP3, more details will be provided in the deliverables of WP2 and WP3. Through discussions within WP1 and the analysis of the use cases of each node, some key requirements for the platform from the perspective of WP1 concerning the data flow could be formulated. The following requirements should be understood as subjects of discussion that will be forwarded to WP2 and WP3:

- The AI-EFFECT platform should provide TEF users with access to the TEF. For this, an authentication process for the TEF users is needed. The platform should provide access to an onboarding process for new TEF users, as well as access to clear documentation on how the TEF can be used.
- The TEF users can request TEF Services through the platform. The choice of services should be guided by the TEF Test Use Cases that have been set up by the nodes, which represent a pipeline of TEF Services.
- For the execution of TEF Services, the exchange of AI tools, training data and other necessary data needs to be facilitated. Also, requirements to the characteristics and scope of the AI tool for using the TEF Services need to be accessible to the TEF user.
- The results of TEF Services should be returned to the platform user - potentially with some harmonized visualization.

Considering these key requirements, two platform design scenarios are presented. The platform scenarios are not exhaustive, but present two extremes, and are created to support the discussion between WP1, WP2, and WP3 about the role of the platform. In the “lean platform” scenario, the platform only facilitates the key requirements described above, meaning the interaction between the TEF user and the TEF. In the “holistic platform” scenario, the platform covers not only the interaction between the TEF user and the TEF, but also other intranodal, internodal, and potentially intra-stakeholder data sharing. This may induce a more modular TEF structure but may require a high adaptability of the platform. The system boundaries of the platform in the two scenarios are visualized with the dotted red lines in **Error! Reference source not found.**.

If the data sharing channels are not managed by the platform, they are established by the stakeholders directly, who are responsible for both the technical implementation of data exchange systems and the legal frameworks that enable them.

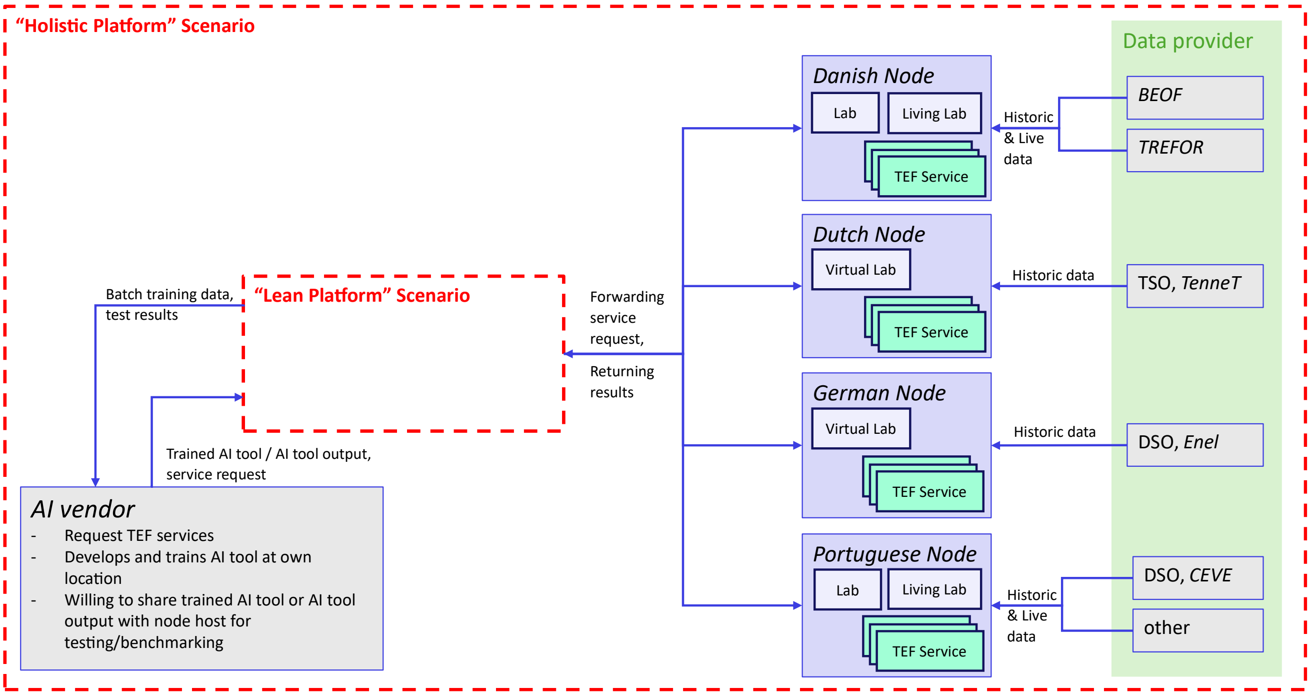


Figure 8: System boundaries of the AI-EFFECT platform for the two scenarios

3.4 Interoperability

Interoperability is instrumental to the AI-EFFECT project in several ways: in the context of data sharing technologies, but also on a functional level. While the functional interoperability is ensured to a certain level by the modular structure of the TEF, including the shared TEF Services, applying TEF Services for use cases they were not developed for may require some adaptation.

In the context of data sharing, interoperability must be ensured on an intranodal, internodal and on the intra-stakeholder level. Internally in each node, the test and experimentation facilities must be interoperable with the node itself, which facilitates the interaction with the TEF users through the AI-EFFECT platform. Additionally, AI vendors may want to test AI solutions on facilities from multiple nodes, which requires that the nodes are interoperable. Finally, some TEF Test Use Cases involve interoperability between stakeholders, e.g., energy providers and flexible consumers that can shift their demand in response to transmission/distribution-related problems. The following types of interoperability in data sharing should be addressed:

- Interoperability to share sensitive data (e.g., anonymization, aggregation, synthetization) and confidential AI tools (e.g., encryption)
- Syntactic interoperability, achieved through, e.g., common data models
- Semantic interoperability, achieved through, e.g., ontologies
- Technical interoperability, achieved through, e.g., APIs, containerization, co-simulation tools

In the AI-EFFECT project, the use case design and node analysis in Task 1.1 and the operating principles developed in Task 1.2 (and disseminated in Deliverable 1.2) will inform the collection of interoperability mechanisms in WP2 (e.g., in Task 2.1). The selection and usage of these mechanisms will be informed by the minimum interoperability mechanism (MIM) framework developed by Open & Agile Smart Cities & Communities (OASC), which involves the identification of strictly necessary requirements for interoperability and the identification of technology-based mechanisms that can satisfy the requirements.

4 Conclusion

Work Package WP1, and specifically Task T1.1, establishes the foundation of the AI-EFFECT project. Its primary objective is to design the nodes and TEF Services from which the fundamental structure of the TEF and requirements for the platform are developed and refined in subsequent work packages WP2 and WP3.

There are two main outcomes of T1.1: First, the definition of the TEF Services, which pose a minimal set of services that will be realized within the AI-EFFECT project. Second, the definition of the TEF Test Use Cases and the corresponding Business Use Cases, documented by the first two sections of the TEF Test Use Case templates based on the HE AI4REALNET MS Word template adapted from ISO/IEC TR 24030. This definition of the TEF Services and use cases relates closely to the project objective PO1, and the expected result ER2 of the grant agreement, as the nodes structure and use cases were specifically chosen to be strategically relevant to European utilities and consumers. Furthermore, the TEF Test Use Case descriptions specify human-AI interaction (Dutch node) and testing requirements, end-to-end certification from simulations to the living labs (Danish node), and verification (Danish node), as specified in the grant agreement.

The TEF Test Use Cases include:

- Multi-energy and sector coupling (Danish node)
- Congestion management within transmission systems (Dutch node)
- Distribution Network Congestion Management for Renewable Integration (German node)
- Performance evaluation of AI-powered Virtual Energy Manager (Portuguese node)
- Performance evaluation of energy-sharing mechanism (Portuguese node)
- Testing of DER scheduling/control algorithms (Portuguese node)
- Validation of Load and RES forecasting algorithms (Portuguese node)
- Performance evaluation of network monitoring algorithms (Portuguese node)

The specific role and design of the digital platform, as well as the exact implementation of the use cases and nodes are still subjects of discussion. Thus, this report provides the current state of the discussions at WP1 to WP2 and WP3. To bridge the gap between high-level node designs and concrete platform requirements, a more detailed specification of workflows for each TEF Service is necessary. These specifications should clearly describe the associated data flows and will be developed in the next step by the partners. The resulting outputs will include sequence diagrams illustrating the exchange of data on the internodal, intranodal and intra-stakeholder level. As these workflow descriptions are still under development, they are not included in this report but will be included in later deliverables.

5 Appendix

5.1 TEF Service list

Table 3: TEF Service list containing the services of all four nodes

#	TEF Service	Brief Description	Input Information	Output
1	**Data provision**	A data provision service refers to a service that supplies, manages, or facilitates access to data for AI-tools like training, testing, and validating in a local or distributed manner. This can include various types of data and access, such as databases, APIs, or other storage solutions. Providing data directly from a simulation or an experiment is also possible. Data can be provided as complete data set at once or as data stream over time. Data provision might include the selection of a communication protocol. The primary goal of a data provision service is to ensure that the necessary data is available, reliable, and accessible in a format that meets the needs of the AI-tool. It may also involve processes like data collection, processing, storage, and distribution. VILLASnode might be an enabler for data provision.	Assuming VILLASnode is used: configuration file specifying origin and destination of the required data.	Dataset incl. metadata
2	**Knowledge Store**	Providing dataset augmentation with additional relevant features for AI model training.	Request, dataset	Dataset incl. additional features
3	**Data synthetization**	This service provides synthetized grid data for the users. The synthetized data is intended to be used to develop AI-based tools in a variety of simulated grid environments. The synthetized data includes information about grid connectivity, substation configuration, electrical circuit parameters of the grid, and injections (generation and load) for one or several time steps. The user supplies data to the synthesizer, either as (a range of) physical grid parameters or a representative set of historical data, and a mode that specifies the synthetization. The service applies	The user supplies data to the synthesizer, either as (a range of) physical grid parameters or a representative set of historical data, and a mode that specifies the synthetization).	Dataset incl. metadata.

		specified algorithms based on the mode to synthesize data. The user receives back a dataset. *Disclaimer: In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.*		
4	**Benchmarking**	This service benchmarks a developed AI-based method against state-of-the-art methodologies and/or basic methodologies. The AI-based method will be compared directly to a chosen methodology or a ranking list. The service applies metrics, test cases, and a controlled environment (hardware and software) to benchmark.	AI tool, training & testing data used for benchmarking, benchmark case/ application selection, benchmark algorithm selection, evaluation metrics selection	Performance metrics compared to state-of-the-art tool, basic tool, or ranking list
5	**Explainability Analysis**	This service provides clear and coherent explanations for single prediction or a set of predictions. By obtaining the shapley values for a trained AI tool, this service estimates the importance of different features to arrive at the predictions. It provides an estimate on how individual features and combinations of features relate to the specific predictions.	AI tool as a black box, test dataset, depending on the type of AI tool used (confidentiality?), list of features, selection of feature combinations for analysis	Feature importance table / KPI table, raw data table, script with plots that reads raw data table and provides explanations
6	**Environment for Mathematical Verification**	Service that generates a suitable verification environment for a specified AI tool. This service can be understood as a wrapper to assess if verification can be provided for the specific AI tool and the business test case. It further translates the request into the verification setup that is the required input for the mathematical verification. If verification is not possible for the specific AI tool, the adaptation/development of a verification algorithm would require human intervention by the service provider.	Test case, AI tool specification (similar information required as in the TEF Test Use Case templates)	If verification possible: Verification setup (incl. AI class, input domain and constraints) Otherwise: need for verification algorithm adjustment, initiate communication between service provider and TEF user

7	**Mathematical Verification**	The node provides tools for mathematical verification for a given class of AI tools. Different verification algorithms are required for different classes of AI tools considering the type of the AI tool (neural network, LSTM, etc.) and its application (optimization, forecasting, control, classification, etc.). For example, for optimization problems, rigorous performance guarantees against user-established constraints or distance to optimality are provided for a user-defined input domain. For classification and forecasting tools, verification may detect abnormal outputs and validate the occurrence of unfeasible or unrealistic outcomes. Mathematical verification may be used before simulation and lab-based testing, to identify critical operating regimes for the AI tool.	Trained AI Tool with the complete structure and parameter values (white-box AI tool), Verification setup incl. constraints, AI class selection, Input domain for verification, selection of evaluation metrics	Verification results for the selected evaluation metrics, for example worst case constraint violations and distance to optimality
8	**Simulation-based Testing**	This service tests AI tools using virtual environments based on predefined experimental designs. It executes various operational scenarios to evaluate the tool's performance under realistic conditions. The system models used in these simulations may be validated against lab measurements. The output includes the performance metrics as per the experimental design.	Experimental design specification, AI tool, access accreditation	Test results (e.g., simulation traces, performance metrics)
9	**Experimental Design for Simulation-based testing**	This service defines the structure and parameters for testing AI tools within virtual environments based on user specifications. It includes the development of operational scenarios, test cases, and evaluation and validation criteria. The experimental plan ensures reproducibility and traceability of results.	Test case, AI tool specification, application domain, access accreditation	Experimental design, assessment criteria
10	**Lab-based Testing**	This service conducts testing of AI tools within physical laboratory environments. The AI tool is integrated into operational setups corresponding to the experimental design and is subjected to actual input conditions. This testing provides empirical validation and performance metrics that can be compared with simulation-based results, where applicable. It can also include a hybrid approach with digital twins.	Experimental design specification, AI tool, access accreditation	Test results (e.g., measurements, performance metrics)

11	**Experimental Design for Lab-based testing**	This service defines the structure and parameters for physical testing of AI tools within controlled laboratory settings. It specifies the setup requirements, instrumentation, input conditions, and data collection methods. The experimental plan ensures reproducibility and traceability of results.	Test case, AI tool specification, application domain, access accreditation	Experimental design, assessment criteria
12	**Living-Lab Testing**	This service enables testing of AI tools in real-world environments involving actual users and infrastructure. This approach captures the AI tool's behavior under dynamic and often multi-stakeholder conditions. It offers insights into usability and robustness in live scenarios, complementing the controlled lab- and simulation-based testing.	AI tool, access accreditation	Test results (e.g., measurements, performance metrics)
13	**Human-AI Interaction Testing**	This service will test the human-AI interaction enabled by a user interface between an AI tool and a human system-operator. Metrics and test cases will systematically assess quality of human-AI interaction. Depending on the use case, testing may happen in the form of expert interviews.	AI tool with user interface, experimental design specifications	Test results (e.g., performance metrics)
15	**Experimental Design for Human-AI Interaction Testing**	This service defines the structure and parameters for physical testing of AI tools within controlled laboratory settings. It specifies the setup requirements, instrumentation, input conditions, and data collection methods. The experimental design ensures reproducibility and traceability of results.	Test case, AI tool specification, application domain	Experimental design, assessment criteria

5.2 TEF Test Use Case Template

Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	***Application(s)***	***Name of Use Case***
UC.X		

1.2 Version management

Version Management			
Version No.	***Date***	***Name of Author(s)***	***Changes***
0.1	DD.MM.YYYY		

1.3 Business Use Case(s)

Priority	***Business Use Case***	***Scope Description of the Business Use Case***
1	Name Business Use Case	This describes the Business Use Case; the application of AI to a process, a challenge or issue in the energy sector. Example: TSO operational planning Text

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning Text
Objective(s)	The use case intention; what is to be accomplished; who/what would benefit. Text

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview. 150 words max Text
Complete description
Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. Text
Stakeholders
Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. Text
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. Text
System's threats and vulnerabilities
Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. Text

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	TEF Service	Objective of the service applied to the TEF Test Use Case
1	Example: Data Synthetization	Example: Synthetize grid operation data to ensure privacy to data provider.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	Use case description (short)

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	***TEF service description (short)***

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Text, Example: Data Synthetization		
	Name	***Description***
KPIs TEF Service implementation	Example: Data privacy/security Text	KPIs evaluating the performance of the TEF Service implementation. Example: Original data cannot be reverse-engineered from synthetic data. Text
KPIs TEF Service & Business Use Case combination	Example: Variety of synthetized TSO grid data (business use case X) Text	KPIs evaluating a specific business use case TEF service combination, e.g. the output of the specific TEF service for a business use case. Example: The synthetized data shows a variety of X grid connectivity, substation reconfiguration and injection combinations that lead to real congestion scenarios. Text

TEF Service: Text, Example: Explainability Analysis		
	Name	***Description***
KPIs TEF Service implementat	Example: Service output calculated Text	Example: The defined KPIs below are successfully computed and forwarded to the user Text
KPIs TEF Service & Business Use Case combination	Example: Shapley values for each feature (business use case X) Text	Example: Shapley value for each input feature to the forecasting algorithm is obtained Text

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate for the functionality of the TEF. Example: End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints). 3) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign. Text
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the TEF Service implementation KPI from above. Text
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the TEF Service implementation KPI from above. Text

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Example: Data privacy/security (data synthetization) Text	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Run the data synthetization on historical data and show that original data cannot be reverse-engineered. Text

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others. Text
Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. Text

2.5 Standardization opportunities and requirements

Classification Information

Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/ https://www.etsi.org/committee/1640-sai Text
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. Text

2.6 Challenges and issues

Challenges	***Mitigation***
Descriptions of challenges and issues of the TEF Test Use Case. Text	Explain how these challenges are addressed or mitigated. Text

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. Text
Scalability	Discuss how the use case can be scaled for future needs. Text

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case Text
Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SGD that are within the scope of this use case. Text

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
		report, mandates and regulatory constraints, paper, patent, press release	Public / confidential	Where does the document influence the use case?		

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[1]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	*Tasks*	*Requirements for Data Exchange*
DSO	Data provider	Data cannot be exchanged directly because of security
Institute	AI development	None

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	*Description of the step*
Source	- Partner x provides data from a data base (historic data), needs synthetization - Partner y provides the output of AI tool
Time resolution	Samples of the data base (historic data) are read every 1 s
Destination	- Partner y receives the data to train the AI tool - Partner z receives data to parse in a simulation (time constraint of the simulation, every 50 ms new setpoint)

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	*Use*

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

Requirement description

[1] VILLASnode | VILLASframework

Programming language	
Input (data format)	
Output (data format)	
Interface	Protocol? API? Text
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Text
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Text
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Text
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Text
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Text
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Text
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Text
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? Text

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***
Runtime	Runtime of the AI algorithm should not exceed X days

3.6 Requirements for Data Synthetization

Usage of the TEF service “Data Synthetization” [2] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

	Requirement description
Need for synthetization	Data privacy/security, increased variety of data for AI tool training/testing? Text

[2] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

Type of data	Grid connectivity, substation configuration, circuit parameters or injection data? Text
Time details of data	Time series or snapshot data, minute/hourly/daily/… resolution, time horizon of one day/month/year/… Text
Location of data synthetization	Where should data synthetization take place? Could the data be sent to the Dutch node to synthetize it there or not? Text
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	***Requirement description***
Data manipulation	Offset calculation, alignment of timestamps, …

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	***Requirement description***
GPU capacity	Provide GPU capacity for training AI algorithm
Virtual machine	Hosting node model, simulation and possibly VillasNode (therefore Linux)

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	***Requirement description***
RTDS data	Real time data exchange

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	***Task: Description***	***Objective***

	Data set: Get different data	Test AI algorithm with new data
	Real-time simulator: calculated setpoints by AI algorithm on a simulated grid	Verify that AI algorithm can cope with dynamic inputs/outputs

5.3 TEF Test Use Case Descriptions

5.3.1 Danish Node

Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF (Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	*Application(s)*	*Name of Use Case*
UC.X		Multi-energy and sector coupling

1.2 Version management

Version Management			
Version No.	*Date*	*Name of Author(s)*	*Changes*
0.1	20.04.2025	Johanna Vorwerk	Copy from previous version
0.2	16.09.2025	Johanna Vorwerk	Updates for final report
0.3	16.09.2025	Paul Bannmüller	Final formatting

1.3 Business Use Case

Priority	*Business Use Case*	*Scope Description of the Business Use Case*
1	District Heating Load Forecasting	This use case focuses on predicting the energy consumption for heating in a district heating grid using ARMAX models. Inputs include numerical weather forecasts,

		local temperature measurements, and time-related data. The goal is to improve the accuracy of heat load forecasts to optimize the operation of heating plants.
1	Spatial Hierarchies for District Heating Load Forecasting	The primary business use case supports district heating system operators in the control of their plants through AI-powered algorithms. The primary goal is to predict district heating loads using machine learning models and control the operation of heating plants. This process involves several critical services that require the robustness and reliability of applied AI solutions. The system setup: The district heating grid consists of a production plant and a heating pipe (potentially with heat exchangers in between) that sequentially heats up the home of each consumer. The return water from the consumers travels through a separate pipe back to the production plant (potentially through a heat exchanger). The temperature is measured at a set of critical (i.e., representative) points in the grid in order to ensure that the consumers receive sufficiently hot water. The developed AI Tools will control the temperature for each customer and flow velocity in the network. With improved controllers operation can be optimized, system losses are reduced and hence cost reductions are expected. This use case involves predicting the district heating load for different subnets connected to a district heating plant. The focus is on using time-series and graph-based machine learning models to predict hourly DH load up to 48 hours ahead, considering spatial information in the predictions.
1	Forecasting of PV Arrays	The aim is to forecast the electricity produced by PV arrays. Inputs include weather data and PV production data.
2	District Heating Temperature and Flow Velocity Control	This use case involves controlling the temperature and flow velocity in a district heating system using generalized model predictive control and ARMAX models. The goal is to ensure consumers receive sufficient hot water by adjusting supply temperature and flow velocity based on temperature measurements at critical points in the grid.

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning End-to-end testing and validation of AI-Tools that enable and improve sector coupling of district heating and power systems. The developed testing pipeline supports AI tool training, enhances explainability, performs verification, and permits testing in controlled laboratory and living laboratory environments.
Objective(s)	The use case intention: what is to be accomplished; who/what would benefit. 1. Develop an environment for explainability analysis and mathematical verification where AI developers obtain insights and can obtain mathematical performance guarantees for their AI-tools. 2. Develop a validation procedure where developers can conduct lab-based of AI-tools for the operation of multi-energy systems including district heating plants and power systems at distribution level. 3. Develop a validation procedure where developers can conduct open-loop living-lab-based testing of AI-tools for the operation of district heating systems and compare their setpoints with real setpoints in real time.

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview.150 words max This TEF Test Use Case supports AI developers with the training, explainability, verification and testing of advanced AI-tools that enhance the operation of multi-energy systems. In detail, the TEF Test Use Case permits explainability analysis, mathematical verification, laboratory-based and living-laboratory-based testing to ensure robust and reliable AI solutions, enhancing efficiency, sustainability and trustworthiness of AI methods in multi-energy systems. Laboratory testing is conducted in Syslab at DTU facilities and permits controlled operation of an integrated energy system including a power system and district heating units. Living lab tests are conducted in an open-loop manner using actual district heating consumer data from Denmark.
Complete description
Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. This TEF Test Use Case aims to enhance the operation of multi-energy systems using AI tools. It provides an end-to-end validation chain for AI tools in the domain. Users will begin with explainability analysis to ensure AI model predictions and decisions are understandable and trustworthy. This fosters confidence in the AI solutions. Next, mathematical verification provides rigorous performance guarantees, validating AI models against user-defined constraints to detect abnormal outputs. This ensures AI models meet specified performance criteria and produce realistic outcomes. Lab-based testing follows, where AI models are tested under diverse operating conditions to ensure robustness and reliability. This step confirms that AI models perform consistently and accurately across different scenarios. With the integration of AI models into a lab framework for real-world testing, their performance is validated under actual operating conditions. This comprehensive approach ensures AI solutions are reliable and effective in practical applications. For applications in district heating systems, living-lab testing follows as a final stage. This testing is conducted in an open loop, i.e. the AI tools cannot control setpoints in the true system. However, it permits to compare computed setpoints and targets with the actual system operation. The TEF will connect the real-world district heating system at SysLab, DTU, enabling comprehensive testing and validation of AI models. This integration ensures that AI solutions are tested under realistic conditions, enhancing their reliability and effectiveness in practical applications. The living lab tests assess the AI tools behavior based on data from real consumers. Specifically, hourly measurements of heat demands from 6000 district heating consumers is enriched with weather forecasts (air temperature, wind speed and direction, solar radiation), which allows the evaluation of heat load forecasting accuracy and an assessment of the central supply temperature and flow velocity (e.g., at a substation) resulting from the heat load forecasts and the supply temperature measurements at the consumers. However, as the data comes from real consumers, the central supply temperature and velocity are not implemented in practice. Together with the tests at SysLab, DTU, the living lab tests provide a picture of the AI solutions behavior under realistic operating conditions and in response to real data.
Stakeholders
Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. District heating operators, DSOs, private, commercial, and industrial district heating customers
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. Competitiveness of heating supply, safety and trustworthiness of AI tool integration, consumer heat demand data.
System's threats and vulnerabilities

Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. Incorrectly applied AI solutions may prioritize the safety of supply of certain customers, e.g., based on demography, unsafe data handling in the AI solutions might inadvertently share sensitive information about heat demand.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	*TEF Service*	*Objective of the service applied to the TEF Test Use Case*
1	Explainability Analysis	This service provides clear and coherent explanations for single prediction or a set of predictions. By obtaining the shapley values for a trained AI Tool, this service estimates the importance of different features to arrive at the predictions. It provides an estimate on how individual features and combinations of features relate to the specific predictions.
2	Mathematical Verification	The node provides tools for mathematical verification for a given class of AI tools. Different verification algorithms are required for different classes of AI tools considering the type of the AI tool (neural network, LSTM, etc.) and its application (optimization, forecasting, control, classification, etc.). For example, for optimization problems, rigorous performance guarantees against user-established constraints or distance to optimality are provided for a user-defined input domain. For classification and forecasting tools, verification may detect abnormal outputs and validate the occurrence of unfeasible or unrealistic outcomes. Mathematical verification may be used before simulation and lab-based testing, to identify critical operating regimes for the AI tool.
3	Laboratory-Based Testing	This service conducts testing of AI tools within physical laboratory environments. The AI tool is integrated into operational setups corresponding to the experimental design and is subjected to actual input conditions. This testing provides empirical validation and performance metrics that can be compared with simulation-based results, where applicable. It can also include a hybrid approach with digital twins.
4	Living-Lab Testing	This service enables testing of AI tools in real-world environments involving actual users and infrastructure. This approach captures the AI tool's behavior under dynamic and often multi-stakeholder conditions. It offers insights into usability and robustness in live scenarios, complementing the controlled lab- and simulation-based testing.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
Testing and validation of AI-based tool e.g., “District Heating Temperature and Flow Velocity Control”	Evaluate the performance of the AI tool in different conditions and critical points by real-time experiment and check the expected results with real-time observation.

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
Co-Optimization of PV Arrays and Electric Boilers	This use case aims to optimize the use of electricity produced by PV arrays within a heating plant. The goal is to use a PV forecast and the excess electricity to run electric boilers, reducing the need for burning biomass. Inputs include weather data and PV production data.

Optimization of District Heating Plant Operations	This use case aims to optimize the operations of a district heating plant to improve heat production efficiency. The focus is on continually optimizing set-points based on current operational conditions using data-driven approaches, potentially leveraging Physics-Informed Neural Networks (PINNs).
Demand-side flexibility exploitation	The objective of this use case is to shift the heating demand through dynamic prices. During periods where it is expensive to operate the district heating plant, a high price is used to incentivize the customers to shift their demand to periods where it is less expensive. AI tools can both be used to predict how consumers will respond to prices and to compute prices that are expected to result in a given desired shift in the heat demand.

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	***TEF service description (short)***
Data Provision	This service provides access to an existing data set that can be employed for developing AI tools
Benchmarking	This service benchmarks a developed AI-based method against state-of-the-art methodologies and/or basic methodologies. The AI-based method will be compared directly to a chosen methodology or a ranking list. The service applies metrics, test cases, and a controlled environment (hardware and software) to benchmark.
Simulation-based testing	This service tests AI tools using virtual environments based on predefined experimental designs. It executes various operational scenarios to evaluate the tool's performance under realistic conditions. The system models used in these simulations may be validated against lab measurements. The output includes the performance metrics as per the experimental design.

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

Use case and service demonstration for the Danish node and their prioritization (1 is highest).

<table>
<tr><th></th><th>Explainability Analysis</th><th>Mathematical Verification</th><th>Lab-Based Testing</th><th>Living Lab Testing</th></tr>
<tr><td>District heating forecasting</td><td rowspan="2">1
(similar methods)
Responsibilities:
Service: Bastien, Spyros, Johanna
AI-Tool: DTU Compute, BEOF
FMU/FMI: Magnus implements first FMU, with DTU Wind input
FIRST DEMONSTRATOR</td><td rowspan="2">6
(similar methods)
Responsibilities:
Service: Bastien, Spyros, Johanna
AI-Tool: DTU Compute, BEOF</td><td></td><td>5
Responsibilities:
Service: Tobias, Magnus?
AI-Tool: Tobias, Magnus</td></tr>
<tr><td>Generation forecasting</td><td></td><td></td></tr>
<tr><td>District heating temperature and flow velocity control of DH plants</td><td>2
Responsibilities:
Service: Bastien, Spyros, Johanna
AI-Tool: Tobias, Magnus, potentially Bastien & Johanna</td><td>3
Responsibilities:
Service: Bastien, Spyros, Johanna
AI-Tool: Tobias, Magnus, potentially Bastien & Johanna</td><td>4
Responsibilities:
Service: Henrik, Zahra, Kai
AI-Tool: Tobias, Magnus, potentially Bastien & Johanna</td><td>5
Responsibilities:
Service: Tobias, Magnus?
AI-Tool: Tobias, Magnus, potentially Bastien & Johanna</td></tr>
</table>

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.

Please fill in a copy of the table for each TEF Service.

TEF Service: Explainability Analysis		
	Name	**Description**
KPIs TEF Service implementation	Shapley value data table delivered	Data table with Shapley values is available and successfully delivered to the user
	Script (applies to all business use cases)	A script that presents and explains the Shapley value analysis is delivered to the customer.
KPIs TEF Service & Business Use Case combination	Shapley values (applied to all business use cases)	Shapley values, including feature interactions, are computed and stored in a data table. These values quantify the contribution of each input feature—and feature pair—to the model's prediction.
	Accuracy (applied to all business use cases)	User-selected standard performance metrics are obtained (Root mean squared error (RMSE), relative RMSE (RRMSE), mean absolute error (MAE))

TEF Service: Verification		
	Name	**Description**
KPIs TEF Service implementation	Verification results delivered	The implementation delivers the verification results, in case no verification results can be obtained, it delivers and understandable error message and informs the operator.
	Verification setup received	The verification setup required for AI tool verification including all required data is received and checked. The verification setup contains a list of KPIs to be obtained during testing.
KPIs TEF Service & Business Use Case combination		
	Distance to optimality (district heating flow velocity control)	The verification computes the distance from the AI tool control setpoints to the optimal control setpoints.
	Worst case constraint violation (district heating flow velocity control)	The verification computes the worst-case violation of the AI tool with respect to pre-defined operational constraints.
	Extreme forecast (district heating and generation forecasting)	The verification identifies which input conditions lead the AI tool to produce its maximum forecast.
	Min / Max prediction (district heating and generation forecasting)	The verification assesses the maximum and minimum output of the AI Tool.
	Robustness of prediction (district heating and generation forecasting)	The verification identifies the size of the input region over which the output remains within an acceptable range for regression, or within the same class for classification.

TEF Service: Lab-Based Testing		
	Name	**Description**
KPIs TEF Service implementation	Test Success Rate	Ratio of successfully executed planned test scenarios over the total number of executed tests
	Control Accuracy	Deviation between expected setpoints and actual outputs during testing a controller (e.g., district heating controller)
	Latency	Time delay between sending a control signal and the system response
	Data Logging	Ratio of required data points recorded and stored during test sessions.
	Communication Failure	Ratio of dropped communication packets or failed commands during tests
KPIs TEF Service & Business Use Case combination	Energy spent on heating (control of district heating systems)	The amount of energy spent on heating the water supplied to the buildings. This can both be 1) an estimate based on the output of

		the algorithm and a nominal efficiency and 2) the actual measured heat consumption. Both values can be useful to the AI vendor.
	Energy spent on pumping (control of district heating systems)	The amount of energy spent on pumping the water supplied to the buildings. As for the energy spent on heating, it can both be 1) an estimate based on the output of the algorithm and a nominal demand and 2) the actual measured demand.
	Heat supplied below setpoint (control of district heating systems)	The amount of energy that was not supplied despite the setpoint being known by the AI algorithm, i.e., in which controls intervals did the algorithm fail to supply the amount of heat and by how much.
	Heat supplied above setpoint (control of district heating systems)	The amount of energy supplied in addition to the prescribed demand. This provides the AI vendor with an estimate of the potential for improvement.

TEF Service: Living-Lab Testing		
	Name	***Description***
KPIs TEF Service implementation	Resolution	The time series data should be on an hourly basis.
	Time series data sets do not have missing values (imputation)	Some AI solutions may not be able to mitigate missing data in the time series. Therefore, the data should be imputed (i.e., missing values should be filled out) or the data sets should be split into separate sets.
	Aggregation	The time series data sets should be aggregated into a representative set of clusters of consumers/meters.
	Enrichment	The heat load time series data should be time-stamped and enriched with weather forecasts (air temperature, wind speed and direction, and solar radiation).
	Data/privacy	The aggregation of the time series should be sufficiently coarse that it is not possible to identify individual consumers, e.g., each cluster should contain at least 10 consumers.
KPIs TEF Service & Business Use Case combination	Forecasting accuracy (conventional)	The forecasting error is reported in terms of typical quantities: 1. Root mean squared error (RMSE), 2. relative RMSE (RRMSE), and 3. mean absolute error (MAE).
	Forecasting accuracy (failure to supply)	The forecasting error is reported in terms of insufficient delivery of heat: 1. Number of hours where the customer was unable to meet their demand, 2. the amount of energy not delivered (based on an assumption of the minimum consumer return temperature), and 3. the average amount of energy not delivered per hour.
	Heating and pumping costs	Based on data from the real-life consumers, an AI solution may determine optimal temperature and flow velocities. Each of these can be associated with a price, and the average cost of operation can be assessed on a per-hour basis. (However, since the actions are not effectuated, this will only be an indicator of the types of decisions the algorithm will make.)

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate in the TEF. End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Explainability analysis enhances helps users to identify important features and explain the importance of features to arrive at specific predictions. 3) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints), how far it is from optimality (where possible) and otherwise assesses under which conditions extreme predictions are obtained. 4) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign.
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the KPI from above. The TEF certification pipeline can guarantee that the AI tool will not violate critical operating bounds or constraints for a specific operating region (an output that the AI tool is not guaranteed to not violate is also acceptable) The lab tests confirm the mathematical guarantees and further extend the verification considering actual operating regimes where measurement noise, and communication delays are present. If the mathematical guarantees cannot be confirmed, then there is a clear justification about the aspects that the model is impossible to capture.
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the KPI from above. The mathematical verification tool cannot extract a result either because the problem is very difficult to solve (i.e. it is impossible to capture the AI Tool in a set of equations for optimization) or because it is impossible to model the underlying system sufficiently well. No connection of the results between the mathematical verification and the lab testing. Properties that have been verified through the mathematical verification obtain an opposite result in the lab testing. This means a misalignment between the mathematical models and the actual components in the lab. Or a misalignment between the data that represent the whole operating region and the actual operating region of the

2.3 Experiment demonstrating the TEF Test Use Case

KPI Name	***Experiment workflow***
Explainability Analysis testing (district heating forecasting)	1. Pack district heating forecast within FMU and sent to DTU Wind via FMI 2. Use FMU to conduct automated explainability analysis, store explainability results in KPI table, auto-generate script for result plots 3. After completion, share KPI table and results with DTU Compute
Verification testing	1. Use interface to prepare verification setup, select KPIs and validation metrics 2. Receive verification setup automatically after completion 3. Send verification setup and white-box AI tool in protected manner to conduct verification services, alternatively, receive tailored verification pipeline for AI tool / use case coordination 4. Conduct mathematical verification, store results in KPI table, automatically generate script to plot results 5. Send results back to user
Two-level district heat flow and temperature controller	6. Define heat demand time series (constant or time-varying) for each control interval, e.g., of length 15 min.

	7. Determine approximate time delays from the heat supply to the consumers. 8. For each control interval 1. measure the supply temperature at the consumers, and 2. let the AI solution determine the central supply temperature and flow velocity based on the measured temperature at the consumers and the known/predetermined heat demands. 9. Evaluate whether the supply temperature at the consumers is sufficient to meet their heat demands (e.g., based on a minimum consumer return temperature)
Co-optimization of PV and Electric Boilers	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Calculate operational cost for TSO from the output of AI tool and compare it with the benchmark model. 1. Define the permissible operating region of PV and Electric Boilers. 2. Consider the region of all possible inputs. 3. Verify that the AI tool has implemented all safety nets to avoid the violation of any operating constraints for all possible inputs. 4. If this cannot be verified, determine the minimum operating region where no violation can be guaranteed, or otherwise suggest measures that can lead to guarantees. 5. Move the AI tool to the lab. For different level of measurement noise, input noise, and delays, assess the performance of the AI tool. 10. If the AI tool performs in unexpected ways, suggest ways of how this can be avoided.

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others. Text BEOF: Inference, prediction, interactivity, recommendation DTU Compute: General stochastic predictive control (incl. Kalman filter), closed-loop real-time optimization (RTO) based on brute force grid search.
Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. BEOF: Azure Databricks DTU Compute: Customized implementation in Python.

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information:

https://www.iso.org/committee/6794475/x/catalogue/
https://www.etsi.org/committee/1640-sai

1. Services: Data Provision and Collection

Standards:

ISO/IEC 42001:2023
Information technology — Artificial intelligence — Management system

ISO/IEC TR 27563:2023
Security and privacy in artificial intelligence use cases — Best practices

2. Service: Data Synthetization

Standard:

ISO/IEC AWI TR 42103
Information technology — Artificial intelligence — Overview of synthetic data in the context of AI systems

3. Service: AI-Tool Benchmarking

Quasi-realistic model for AI development

Standards:

ISO/IEC 25059:2023
Software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality model for AI systems

4. Services: Lab-based testing and Lab-verified model for AI validation

Standard:

ISO/IEC 17025:2017
General requirements for the competence of testing and calibration laboratories

5. Services: Mathematical Verification

Environment for Mathematical Verification
Open AI Test Case Repository
Simulation-based testing

Standards:

ISO/IEC CD TS 42119-3
Artificial intelligence — Testing of AI — Part 3: Verification and validation analysis of AI systems

ISO/IEC 24029-2:2023
Artificial intelligence (AI) — Assessment of the robustness of neural networks — Part 2: Methodology for the use of formal methods

ISO/IEC TR 29119-11:2020
Software and systems engineering — Software testing — Part 11: Guidelines on the testing of AI-based systems

Standardization requirements

Descriptions of standardization opportunities/requirements that are derived from the use case.

- Data models and ontologies for energy system data, specifically district energy data
- Input Data Format – Specification of structured schemas and encoding standards for raw and processed datasets.
- Data Structure for Sharing Among Stakeholders – Definition of interoperable data models and exchange protocols for cross-platform integration.

- Evaluation Metrics – Establishment of standardized performance benchmarks for AI model assessment.
- Data Sharing Transparency and Privacy – Enforcement of secure data governance policies and compliance with regulatory standards.
- Accuracy and Performance Metrics for Different Cases – Development of sector-specific KPIs and validation criteria for different applications.

2.6 Challenges and issues

Challenges	*Mitigation*
Risk of delayed AI models (business use cases) for test and experimentation.	Close collaboration, sharing progress and early communication of delays, fall-back to simpler algorithms if necessary
Co-development or re-use of TEF services between nodes requires synthesized data or a challenging data sharing agreement.	Only synthesized or aggregated data is left on each TEF node.
Mathematical models of components are not aligned with the lab components	Continuous tests
Misalignment of verification and lab-based testing results	Co-development and testing of the services

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. 1. From verification to explainability: Offer insights about when a forecasting AI tool will deliver forecasts that are considered extreme events. This is very relevant for resilience. In other words, based on XX forecasting tool, what are the conditions that could lead to an extreme event? 2. Extend the verification tool to capture NNs, LSTMs, including pooling layers, convolutional layers, etc. 3. Digital twin construction and validation. Sector-coupling scenarios coupling district heating and power system real-time simulation.
Scalability	Discuss how the use case can be scaled for future needs. 1. Use optimization based on GPUs to scale to larger size AI-based tools. 2. digital (virtual) scale-up of laboratory setting. Virtual energy market integration.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case Data privacy

Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SDGs that are within the scope of this use case. Affordable and clean energy (SDG 7)

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
	Technical Report	Digitalization of District Heating	Public	Heat load forecasting and supply temperature control	Technical University of Denmark	https://orbit.dtu.dk/en/publications/digitalization-of-district-heating
DOI: 10.1016/j.apenergy.2021.116872	Journal paper	Heat load forecasting using adaptive temporal hierarchies	Public	Heat load forecasting	Technical University of Denmark	https://www.sciencedirect.com/science/article/pii/S0306261921003603?via%3Dihub
DOI: 10.1016/j.egyr.2021.08.153	Conference paper	Use of smart meters as feedback for district heating temperature control	Public	Heat supply temperature control	Technical University of Denmark	https://www.sciencedirect.com/science/article/pii/S2352484721007575?via%3Dihub
DOI: 10.1007/978-3-031-10410-7_6	Book chapter	Data-Driven Methods for Efficient Operation of District Heating Systems	Public	Heat load forecasting and supply temperature control	Technical University of Denmark	https://link.springer.com/chapter/10.1007/978-3-031-10410-7_6
DOI: 10.1016/j.enconman.2022.116113	Journal paper	Estimating temperatures in a district heating network using smart meter data	Public	District heating model identification	Technical University of Denmark	https://www.sciencedirect.com/science/article/pii/S0196890422008974?via%3Dihub

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[1]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	***Tasks***	***Requirements for Data Exchange***

Node host (DTU Wind, supported by DTU Compute)	• Provide access to the TEF platform • Provide services to customers • Develop and implement explainability analysis, verification, lab-based and living-lab based testing. (DTU Compute is leading efforts for living-lab testing) • Develop AI algorithms • Verification of AI tools through algorithmic tests and in the lab	• Data sharing agreement between Center Denmark. Data Sharing Agreements with BEOF, CDK, TREFOR. [Some data is provided by TREFOR (utility company), which is not a partner in the project.] • Develop an interface/API where External/Project partners can connect to and access data. • AI-Tools need to be sent in different formats for the different services with potentially different security requirements. • Explainability analysis and lab-based testing only require the tools as a black box, while verification required the AI tool including the entire model structure and parameter values.
Data providers (CDK/ TREFOR, BEOF)	Data service provider and enrichment with relevant data sets required by the AI algorithm, provide data for validation, experimentation etc. for other 3rd party AI developers	Exchange of data to DTU through data agreement. Data for AI effect TEF requires synthetization or aggregation due to GDPR issues.
Platform user, AI vendor (DTU Compute & BEOF)	Develop AI Tools that are tested on the platform	Request services through the AI TEF platform, share the required data and the AI tool (as a black box or in a protected manner as a white box) so that services can be conducted. Receive final results back.
Data handling (Energydata.dk & CDK)	Receives data from data providers and provides it to the node host for service provision.	• BEOF data is channeled through energydata.dk • TREFOR data is channeled through CDK

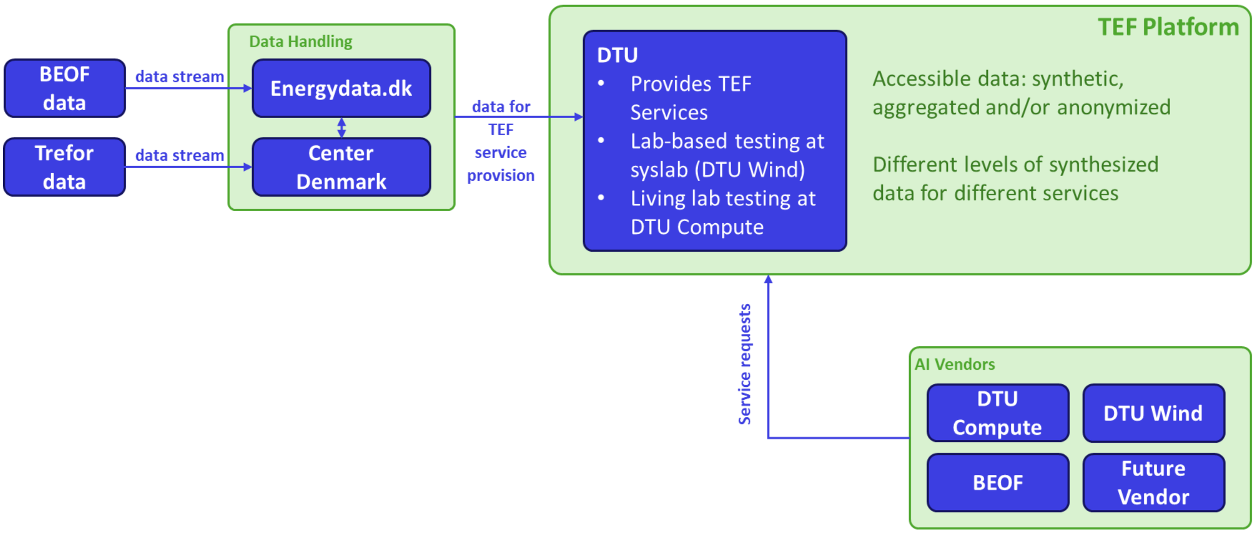


Figure 1: Overview of the Danish node.

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	*Description of the step*
TREFOR Varme and Center Denmark	Center Denmark gathers data from 6,000 of TREFOR Varme's customers in real-time Center Denmark provides access to the data in real-time, e.g., to DTU Compute through an API (historical data can also be provided)
Time resolution	Hourly
Destination	1. DTU Compute receives data, trains/updates model, and computes forecasts/optimal decisions using AI tool 2. DTU Wind receives aggregated data for data provision, explainability and verification service provision.

Step of data flow	*Description of the step*
BEOF	The utility company BEOF provides access to historical data from district heating smart meters. The data will be synthesized (aggregation) before shared with the node. BEOF use their own data to train the developed ai models. However, data for training and testing will be shared with node owner (DTU Wind) for demonstration of various services (use case dependant) BEOF will provide ai model for test and demonstration of services within the node.
Time resolution	Hourly
Destination	DTU Wind (node owner) receive cleaned data from DTU through Azure Delta Sharing protocol. Data receiver (DTU Wind) can freely select tool to utilize data. AI model will be shared using project standard (FMU).

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	*Use*
Available Data Interaction Methods on the Center Denmark Portal:	1. **Download Prefiltered Data** 1. Direct download of specific, prefiltered datasets via the portal interface. 2. **Fetch Data via API** 1. Use **HTTP or Python** clients (like `requests` or `curl`) to interact with the CDK API. 2. Authentication via access token is required. 3. **Post Data to CDK's API**

	1. If the platform allows write access (this should be confirmed in documentation), data can be submitted (posted) to the API endpoints using authenticated requests. 4. **Delta Sharing** 1. Use the **Delta Sharing protocol** to access shared datasets. 2. Three main steps: generate token, fetch profile, and use a Delta Sharing client (e.g., with schema and table name). 3. Accessible via **Swagger**, **HTTP**, or **Python**. 5. **Explore & Test Endpoints via Swagger UI** 1. Swagger provides interactive documentation to explore and test the API endpoints. 2. Supports viewing schemas, making calls with authentication, and trying out parameters live. 6. **Automated Data Ingestion & Processing** 1. While not shown explicitly in the screenshot, platforms like this often support scheduled or event-based data pull/push integrations (to be confirmed in full documentation or by support).
SYSLAB	***Interface Option 1 (mature):*** **Python Library Integration (On-Site Testing)** On-site testing of controller prototype by integrating the SYSLAB (Python) library on client machine connected to SYSLAB WiFi or physical network. (Minimum coordination overhead) Protocols: XMLRPC based proprietary library ***Interface Option 2 (proven in earlier projects):*** **VILLAS Interface (Remote Testing)** Remote live testing of a controller prototype by mapping signals to SYSLAB via a VILLAS interface. VILLAS node is set up on client premises and interfaces with client controller code. Requires real-time support and case-specific preparation of SYSLAB interface (high coordination overhead). Protocols: Villas node protocols; usually using WebServices to avoid setting up VPN. ***Interface Option 3 (to be developed):*** **FMU Interface (On-Site FMU Deployment)** On-site deployment of controller FMU by mapping signals to SYSLAB FMU interface. FMU code is sent to SYSLAB, deployed and tested by staff. Protocols: FMI standard, “SYSLAB FMI library”
BEOF *All data is stored inside an inhouse datalake and is utilized through the Azure environment. This means that all data will be shared using the Delta Sharing Protocol*	Delta Sharing Protocol • Data provider generate the needed Delta Lake tables

	• Data provider generate a Delta Sharing Link (unique per recipient) • Data recipient access the shared data using Power BI, Tableau, Apache Spark™, pandas and Java or other tools FMU Interface
EnergyDataDK	EDDK offers streaming and batch data APIs. For streaming, MQQT is used and for upload is done via CSV files. Datasets and Datastreams are configures on web-interface, either by manual setup or CSV upload. Licensing and privacy is managed at Dataset level. Online tool allows data exploration and basic filtering based on Datastream metadata.

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

Tool 1: Forecasting Heat Demand

Requirement description	
Programming language	Python 3.0
Input (data format)	Aggregate historical heat demands, forecasts of air temperature, wind speed, wind direction, solar radiation, and time of day
Output (data format)	Forecasted aggregate heat demands
Interface	Protocol? API? Python scripts
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Should only run locally in node
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning (offline training on data and online real-time training)
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Regression (used for prediction/forecasting)
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods ARX
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time series data
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) N/A (an internally developed custom Python package has been used)
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? The tool solves a prediction/forecasting problem. It involves continuous random variables, a least-squares objective function (offline batch estimation and online recursive estimation), and there are no constraints.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you

	want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? The model prediction errors should be normally distributed, have zero mean, they should not have autocorrelation, and should not be autocorrelated with the regressors.

Tool 2: Supply temperature and velocity control/optimization

Requirement description	
Programming language	Python 3.0
Input (data format)	Forecasted heat demands and ambient temperature, consumer temperature measurements (supply)
Output (data format)	Optimal central supply temperature and velocity (e.g., at substation)
Interface	Protocol? API? Python scripts
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Should only run locally in node
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning (the MPC algorithm adapts online to an unknown disturbance)
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Adaptive state estimation, control, and economic optimization
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Stochastic bilinear state space model (bilinearity between one input variable and the state variables)
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series data and predictions/forecasts (also time series)
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) N/A (custom implementation)
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? The tool solves the two following problem in order to compute the supply temperature and flow velocity, respectively. 1) The supply temperature is the solution to a discrete-time unconstrained optimal control problem with continuous random variables where the objective function is the expected squared deviation. It is solved in a receding-horizon manner and is also described as unconstrained model predictive control (MPC). 2) The supply velocity is the solution to a chance-constrained real-time optimization (RTO) problem with continuous random variables. The objective function is the economy of operation (in terms of heating and pumping costs), and the chance constraints involve the supply temperature at the consumers (i.e., is it possible to deliver the forecasted heat demand). The problem also involves bound constraints on the manipulated inputs.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? We want to verify that the algorithm does not fail to supply hot enough water to meet the consumers' actual demands, i.e., the algorithm should be robust towards heat load forecasting errors and variations in ambient conditions (e.g., ambient temperature).

Tool 3: Co-optimisation of PV arrays and electric boilers (BEOF)

Requirement description	
Programming language	The ML model will be developed in python, leveraging open-source ML libraries like, i.e., scikit-learn.
Input (data format)	Weather data, either measured or forecasted, for a given hour.
Output (data format)	Predicted PV power output for the hour in question.
Interface	Protocol? API? Model inference will be run on a scheduled basis (e.g., every three hours), where we will use MLflow and the Unity Catalog model registry to query the model in a python script and store the predictions to a table, from where they can be used.
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing only intended for the node for utilization of serviced. Tool should only run locally in node.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Supervised learning and/or statistical learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) To be able to predict PV production, such that production in excess of a certain threshold (e.g., curtailment limit) can be used to run electrical boilers and thereby replace burning of biomass. The focus is on predicting average hourly PV production a number of hours ahead.
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Tree-based
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Scikit-learn
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Model solves an optimization problem of co-optimization of running the electrical boiler when feasible. Tool does still have unknown constraints.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? Text

Tool 4: District heating load forecast (BEOF)

Requirement description	
Programming language	The ML model will be developed in python, leveraging open-source ML libraries like, i.e., scikit-learn.
Input (data format)	A time series of weather data, either measured or forecasted, for a given hour and a number of hours leading up to it.
Output (data format)	Table with the predicted heat demand for the targeted district heating plant. Output comes hourly and for 48 hour ahead.
Interface	Protocol? API? Delta Sharing Protocol, Python scripts
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing only intended for the node for utilization of serviced. Tool should only run locally in node.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Supervised learning and/or statistical learning

Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Predicted DH load for the hour in question, optionally with spatial structure in the output(s) – e.g., separate predictions for different subnets, parts of these, etc., reflecting spatial hierarchy.
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Tree-based model
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Scikit-learn
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? To be able to predict the DH load, such that the production at the DHP can be planned ahead to meet this demand with minimal need for buffer capacity and minimal risk of underproduction. The focus is on predicting average hourly net DH load, around 6-48 hours ahead.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? Verification that the demand does not exceed the available stored energy.

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***

3.6 Requirements for Data Synthetization

Usage of the TEF service "Data Synthetization" [2] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

Requirement description	
Need for synthetization	Data privacy/security, increased variety of data for AI tool training/testing? No need for synthetization, we are aiming for aggregation.
Type of data	Grid connectivity, substation configuration, circuit parameters or injection data? Text
Time details of data	Time series or snapshot data, minute/hourly/daily/… resolution, time horizon of one day/month/year/… Text
Location of data synthetization	Where should data synthetization take place? Could the data be sent to the Dutch node to synthetize it there or not? Text
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	*Requirement description*
Data manipulation	Offset calculation, alignment of timestamps, …
Missing values	Missing measurements in heat load and weather forecast time series must be handled, e.g., by imputation (forward fill/persistence model, mean replacement, etc.).
Enrichment	The heat load data should be enriched with weather forecasts of temperature, wind speed, wind direction, solar radiation. This also requires coordination with the aggregation procedure below where both spatial and temporal alignment is necessary.
Aggregation	The data used for heat load predictions (tool 1) and supply temperature and velocity control/optimization (tool 2) should be aggregated into representative clusters.
Outlier detection	Outliers should be detected and handled. Either removed or manipulated. Examples of found outliers. • Extreme outliers (nummerical causes) • Accumulated outliers. If a meter looses connection the accumulated consumption will be sent when connection is established again.

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	*Requirement description*
Interface CDK	See 3.3 protocol
Interface energydata.dk	
SYSLAB, safety constraints/ requirements for testing in syslab	

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	*Requirement description*
SYSLAB	Operates with approximately ~1s sampling rate.
Living lab	In a real scenario, we expect to receive data hourly.

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	*Requirement description*
None	Handled by Center Denmark platform, See description 3.3
SYSLAB	Generally no privacy issues; receiving data from controllers is subject to safety constraints.

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	*Task: Description*	*Objective*

5.3.2 Dutch Node

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	*Application(s)*	*Name of Use Case*
UC.X		Congestion management within transmission systems

1.2 Version management

Version Management			
Version No.	*Date*	*Name of Author(s)*	*Changes*
0.1	13.05.2025	Paul Bannmüller	First draft

0.2	22.07.2025	Paul Bannmüller	First version

1.3 Business Use Case(s)

Priority	Business Use Case	Scope Description of the Business Use Case
1	Decision support for TSO congestion management	This describes the Business Use Case; the application of AI to a process, a challenge or issue in the energy sector. Example: TSO operational planning Support TSOs in the operation of the transmission system with AI-powered algorithms for decision-making in congestion management

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning Testing and benchmarking of AI-based algorithms for congestion management within transmission systems
Objective(s)	The use case intention; what is to be accomplished; who/what would benefit. **1)** Develop an operational controlled environment for testing AI solutions where technology developers can rigorously test and enhance AI algorithms for congestion management, ensuring they are reliable. **2**) Define a hyper-realistic testing environment using synthesized grid data, a simulator of the power grid, and a user interface enabling exploration of AI solutions under a variety of operational scenarios. **3)** Develop a benchmark to compare the AI algorithms against state-of-the-art and/or basic methodologies, based on specific metrics in a controlled environment.

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview. 150 words max The central ambition of this facility is to hone AI-based algorithms for congestion management, a critical function of control rooms. By providing a digital-physical and controlled environment replete with processes for synthetic grid data generation, the facility would allow for rigorous testing and refinement of these algorithms, ensuring they are robust and effective in managing the grid's congestion, e.g., through sequential topological reconfigurations.
Complete description
Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. • TSO data is synthetized by the data synthesizer, creating a dataset of diverse transmission grid congestion cases defined by grid connectivity, substation design, circuit parameters and injections • AI vendor can use the data to train and test its AI algorithm • AI algorithms for congestion management algorithms are deployed in the TEF and benchmarked against a chosen reference algorithm or a ranking list

• Environment of real control room is imitated by a simulator for the transmission grid and a user interface for human interaction • AI-human interaction through a user interface is tested at the TEF, experiments for this are designed accordingly • User can refine the algorithms and user interfaces and benchmark/test again • (User goes back to real control room and deploys the algorithms there, improves the usability and runs them)
Stakeholders
Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. TSOs, also DSOs impacted by the decisions of the TSOs, energy suppliers, power grid users,…
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. Security of power supply, stability of the power grid, trustworthiness, reputation
System's threats and vulnerabilities
Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. Scenarios of generation/load/grid maintenance/contingencies that have not been respected in the training/development of the algorithms or weak performance of the algorithms in general could lead to misleading suggestions for the decision making. This could lead to poor operation of the transmission system by the TSO.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data synthetization	Synthetize grid operation data to ensure privacy to data provider and create data for AI tool training and testing.
2	Benchmarking	Developing a benchmark for the AI tool on congestion management
3	Human-AI interaction testing	Testing of the Human-machine interaction of a decision support tool deployed in the control room.
4	Experimental design for Human-AI interaction testing	Design of the experiments for Human-AI interaction testing and development of a evaluation metrics.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***
Grid topology testing	Testing tools for new grid topologies by using RTDS related to expansion planning

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
Use synthetized data	Using synthetized data provided by the TEF for training of algorithms that are used within the TSO context (real time/day ahead security assessment)
Data synthetization	Data synthetization as a service for other use cases than TSO data
Grid topology expansion planning	Testing the operation of new grid topologies
Corrective control of topology	The corrective control of the topology can later advance operation of system operators and this TEF node can also help with this development
Outage planning optimization	Outage planning optimization respecting dependencies on resources, grid security etc.
Connection request assessment	Assess connection requests to the transmission grid

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*
Mathematical Verification	Mathematical verification of the AI tool for congestion management
Data provision	Provision of synthetic transmission grid, e.g. in the form of a data base.
Experimental design for simulation-based testing	Service that generates a suitable experimental design based on a specification of the testing need
Simulation-based testing	This service provides simulation-based testing of an AI-tool for a given, existing experimental design. As such, the AI-tool is tested under diverse operating conditions. System models used to test the AI tools may be verified through lab testing. In the future, the node could be equipped with a SCADA system and connected to RTDS simulators.

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Data Synthetization		
	Name	***Description***
KPIs TEF Service implementation	Data flow	The service is accessible by the AI vendor and necessary data can be shared by both the AI vendor and the TSO data provider. This includes TSO grid data and/or metadata for synthetization and synthetized data.
KPIs TEF Service & Business Use Case combination	Data synthesizer operation modes	KPIs evaluating the performance of the TEF Service implementation. Data synthesizer is able to operate in two modes: Using metadata as an input to create synthetized data or using original data that should be synthetized.

	Data privacy/security	Original data cannot be reverse-engineered from synthetic data, even if synthetization method and code are public.
	Realism of synthetized TSO grid data	KPIs evaluating a specific business use case TEF service combination, e.g. the output of the specific TEF service for a business use case. The synthetized data is realistic with respect to the grid design and the operation, including realistic congestion scenarios.
	Diversity of synthetized TSO grid data	The synthetized data shows diverse scenarios for grid connectivity, substation design, circuit parameters, and injection. Metrics will be developed during project.
	Similarity of synthetized TSO grid data to original data	Some generated scenarios must be sufficiently close to the original grid data that testing on synthetic data is meaningful. Note that the constraint of data security must hold. Metrics will be developed during project.
	Time	Synthesizer must be suitable for snapshot and dynamic topology reconfiguration.
	Computational efficiency	The developed approach is computationally efficient. Metrics will be developed during project.
	Adaptability to other use cases	The developed approach is adaptable to other use cases that include similar data structures. This could be shown for one other use case.
	Interface	The interface of the data synthesizer enables data-driven automation and meaningful user control.

TEF Service: Benchmarking		
	Name	***Description***
KPIs TEF Service implementation	Benchmark environment	Benchmarking runs on a controlled environment w.r.t. computational resources, simulation environment, and dataset.
	Benchmark test case	The benchmark test case is well defined and designed so that the performance of the AI tool can be evaluated.
	Benchmark comparison	A baseline/state-of-the-art AI tool or ranking list is given to compare the results of the benchmarked AI tool and rank its performance.
	Data flow	The service is accessible by the AI vendor and necessary data can be shared. This includes the trained AI tool and benchmarking results.
KPIs TEF Service & Business Use Case combination	Speed of solution inference/calculation	Time the AI tool takes given the computational resources to infer/calculate the solution to the congestion problem.
	Quality of solution	Quality of the solution to the congestion problem suggested by the AI tool. Metrics will be developed during the project. For example: cumulative overload, timesteps without overload, number of switching actions, N-1 overload,…
	Example KPIs	In the following, example KPIs for the evaluation of the congestion management tool are listed.
	Maximum Line Utilization under normal conditions (n-0)	Highest observed line loading under the base case (n-0), e.g., without any outages or contingency scenarios. This represents the peak stress on the grid in normal operation.
	Maximum Line Utilization under Contingency (n-1)	Maximum line loading across all single-contingency (n-1) scenarios. Indicates the worst-case stress under failure events of a single element.

	Number of Topological Switching Events	Counts the number of time-steps during which at least one topological switching operation (action) was applied to manage congestion.
	Maximum Topological Depth	Maximum number of distinct stations that were manipulated per time-step.
	Maximum Topological Distance	Maximum topological distance, measured in number of distinct manipulated stations, between two consecutive time-steps.
	Longest Consecutive Overload Duration	Maximum number of consecutive time-steps a single grid element is overloaded.

TEF Service: Human-AI Interaction Testing		
	Name	***Description***
KPIs TEF Service implementation	Human-AI Interaction testing environment	Testing environment includes the a user interface which can be accessed by the AI tool.
	Human-AI Interaction test case	The test case is well defined and designed so that the performance of the AI tool can be evaluated. This is done by the TEF service “Experimental Design for Human-AI Interaction Testing”
	Data flow	The service is accessible by the AI vendor and necessary data can be shared. This includes the user interface of the AI tool and the human-AI testing results.
KPIs TEF Service & Business Use Case combination	Quality of the human-AI interaction	The quality of the human-AI interaction is assessed w.r.t. the metrics developed by the TEF service “Experimental Design for Human-AI Interaction Testing”

TEF Service: Experimental Design for Human-AI Interaction Testing		
	Name	***Description***
KPIs TEF Service implementation	Experimental design methodology	The experimental design is dependent on the AI tool but follows a general methodology.
	Evaluating metrics	The evaluation metrics is dependent on the Business Use Case but follows a general methodology.
KPIs TEF Service & Business Use Case combination	Quality of the human-AI interaction	A metrics is developed to evaluate the quality of the human-AI interaction with the AI tool for congestion management. This metrics needs to be developed but could include measuring trust, complexity…
	Example KPIs	In the following, example KPIs for the evaluation of the congestion management tool are listed.
	Frequency of Human Interaction	Count how often the human operator intervenes in AI-generated decisions during operational scenarios. A high frequency may indicate a lack of trust or reliability in the AI tool.
	Human-AI Decision Agreement	Reflects the level of human operators’ agreement with the AI tool’s decisions, rated on a scale (e.g., 0 – 100).

	Understandability of AI Recommendations	Assesses how well the human operator understands the AI's outputs or suggested actions.'
	Satisfaction with AI-Assisted Decision Support	Human operator's subjective satisfaction with how the AI system supports the decision-making process.
	Enhanced Situational Awareness through AI	Evaluates the extent to which AI support improves the human operator's ability to perceive, understand, and anticipate developments in the operational environment. This also includes training of the operator through suggestions that are beyond the operator's experience.
	Operator Trust in AI Decisions	Measures the degree of trust human operators place in AI-generated decisions or recommendations, typically collected through questionnaires or Likert-scale ratings.

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate for the functionality of the TEF. Example: End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints). 3) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign. The TEF is capable of benchmarking the AI tool for congestion management and testing the human-AI interaction. For this, the TEF can synthetize TSO grid data to create a sufficient number of realistic TSO grid data with congestion cases.
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the TEF Service implementation KPI from above. AI tools can be tested as described above, all KPIs on the implementation of the TEF services are fulfilled and the testing KPIs can be obtained.
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the TEF Service implementation KPI from above. AI tools cannot be tested as described above because one of the TEF services did not fulfill the KPIs defined above.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Data privacy/security (data synthetization)	Describe the workflow and the experiment conducted to evaluate that KPI. Run the data synthetization on historical data and show that original data cannot be reverse-engineered. (exact definition for "reverse-engineerable" pending)
Data flow (all services)	The integration of all of the TEF services needs to be tested. For this, the AI vendor and the TSO data provider need to check whether they can send and receive all the data that is needed. This includes TSO grid data or metadata for synthetization, synthetized data, the trained AI tool, benchmarking results, the user interface of the AI tool, human-AI testing results
Benchmark	Ensure a reproducible benchmark by the same tool multiple times and comparing the outcomes.
Human-AI-interaction testing	Ensure reproducibility by testing the same tool multiple times and comparing the outcomes.

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others.

	Decision support for operation, sequential decision making, reinforcement learning, optimization
Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. Grid2Op, PowerFactory?

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/ https://www.etsi.org/committee/1640-sai Data synthetization: ISO/IEC 5259-2:2024 (and others from 5259), ISO/IEC AWI TR 42103 Testing: ISO/IEC AWI TS 42119-2
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. Text

2.6 Challenges and issues

Challenges	***Mitigation***
Descriptions of challenges and issues of the TEF Test Use Case. AI tool and benchmark aim to solve different facets of the congestion problem and therefore are not comparable	Explain how these challenges are addressed or mitigated. AI tool and benchmark should be build based on a common problem formulation. Therefore, explicit problem formulation must be stated.
Sharing of TSO data within the node for demonstration, or, even more difficult, with any other stakeholder	Data is synthetized before being shared, but no regulations for synthetized data sharing yet from TSO
Sharing of AI tool source code	Training locally at the AI vendor, benchmarking online at node. Benchmarking without sharing the source code can follow a challenge-based approach, similar to setups like the L2RPN or Flatland challenges. In this approach, the AI vendor is required to implement a predefined entry point that allows querying the agent for an action based on a given power grid observation. This way, the vendor only needs to submit the trained model (weights) and the entry point implementation, without revealing the full source code. Also for training locally the simulation environment should also be available to “download” or to access separately from the AI vendor.
Access of node from external parties besides node host	Set up cloud with interface to virtual machine of node host?

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. In the future, the testing and validation capabilities of the node can be extended. Therefore, SCADA and RTDS could be connected to the simulation environment to make it more realistic and to also test the human-machine interaction like it would be in an real control room.
Scalability	Discuss how the use case can be scaled for future needs.

	The benchmark model and computational resources should be scalable to various grid topologies and sizes. Also, it should be adaptable (open-source) and other benchmark models should be able to integrate.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case Text
Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SGD that are within the scope of this use case. 7, 9

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
1	Pre-Print	RL2Grid: Benchmarking Reinforcement Learning in Power Grid Operations	Public	Benchmarking (TEF Service)	Enrico Marchesini (MIT) et al.	https://doi.org/10.48550/arXiv.2503.23101
2	Conference Paper	User experience evaluation of an AI-based decision-support tool for power grid congestion management	Public	Human-AI-interaction testing (TEF Service)	Jan Viebahn (TenneT TSO BV) et al.	http://doi.org/10.54941/ahfe1006694

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[1]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

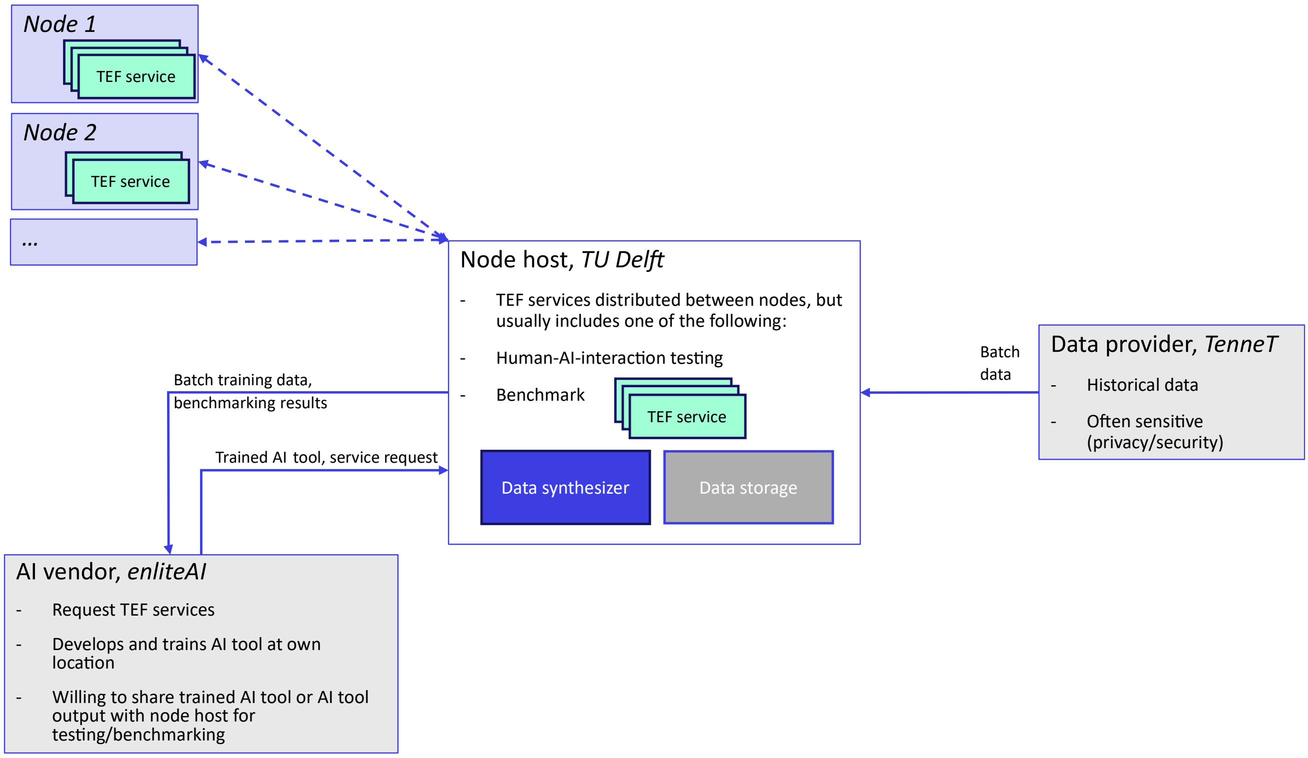
Function of Involved Partners	***Tasks***	***Requirements for Data Exchange***
TSO (TenneT)	Data provider	Data cannot be exchanged directly because of security, must be synthetized before usage. NDA to share data with node host, TSO grid data must be non-reverse-engineerable after synthetization to be used by other parties.

Node host (TU Delft)	TEF node host, provides the infrastructure and the benchmark	IT infrastructure is not directly accessible for others. Need to find a solution for the AI vendor to access for benchmarking and retrieving synthetized data, also for the TSO data provider to send data.
AI vendor (enliteAI)	Trains its AI tool with provided data, tests it against the benchmark and refines the tool	Does not want to share AI tool source code. Trained AI tool (weights) can be shared.

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	*Description of the step*
Source	• TSO provides data from a data base (historic data), needs synthetization • The AI vendor provides its trained AI tool for benchmarking • Node host sends synthetized TSO grid data to AI vendor for training of AI tool, then sends results after benchmarking of AI tool
Time resolution	Internet connection should be sufficient. All the processes run locally (offline) and data is sent/received in batches to then run the processes after.
Destination	• Node host receives data from TSO, synthetizes it, and provides it to the AI vendor locally • The AI provider receives the synthesized data from the node host to train the AI tool on premises. • The node host receives the trained AI tool from the AI vendor to perform benchmarking. • AI vendor receives information about the benchmark testing of its AI tool



3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	***Use***
Internet	All communication between VM and AI vendor, TSO, and AI-EFFECT platform

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

Requirement description	
Programming language	Python, C++
Input (data format)	CSV, XIIDM, JSON
Output (data format)	CSV, JSON
Interface	Protocol? API? Benchmarking without sharing the source code could follow a challenge-based approach, similar to setups like the L2RPN or Flatland challenges. In this approach, the AI vendor is required to implement a predefined entry point that allows querying the agent for an action based on a given power grid observation. This way, the vendor only needs to submit the trained model (weights) and the entry point implementation, without revealing the full source code.

Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing not intended; it should only run on-premises in the node for benchmarking.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Reinforcement learning, optimization (depends on method)
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Control / Sequential Decision-making to mitigate congestion in transmission grids. The control variables are the switches used to reconfigure the network topology (transmission line switches, busbar couplers, substation reconfiguration) and potentially re-dispatch actions.
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Neural network, tree-based, optimization (depends on method)
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Graph-based models, tabular models, optimization models (depends on method)
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Maze-rl, Scikit-learn, TensorFlow, PyTorch. The specific packages used in the implementation are yet to be defined.
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Topology reconfiguration is based on discrete variables for switching couplers, and possible continuous for controlling redispatch. The power flow can be modeled as ACOPF (MIP), DCOPF (MILP) or not explicitly in model-based or model-free RL. The problem is an optimization.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? The goal of a verification could be to ensure that the agent does not take any topological action that is worse (lead to more contingency, destabilize the system, decrease security or similar) than just operating the system in the base topology.

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***
Problem scope	Problem scope and formulation of the tested AI tool must fit the one of the benchmark environment and test case.
Integration into environment	The AI tool should be able to run in a predefined environment which is used for benchmarking.
Computational cost	The AI tool should be capable of running efficiently on a virtual machine with moderately specifications. As a guideline: CPU: ~32 cores GPU: NVIDIA RTX 4090 (or equivalent) RAM: ~250 GiB These specifications may be adjusted in the future.

3.6 Requirements for Data Synthetization

Usage of the TEF service "Data Synthetization" [2] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

Requirement description

Need for synthetization	Data privacy/security, and increased diversity of data for AI tool training/testing. See KPIs for data synthetization in 2.1.
Type of data	Grid connectivity, substation configuration, circuit parameters or injection data
Time details of data	Hourly resolution, day ahead data (horizon of 1 day)
Location of data synthetization	With NDA, the data can be sent to the Dutch node for synthetization
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	*Requirement description*
Data manipulation	Offset calculation, alignment of timestamps, …
None	

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	*Requirement description*
Computational capacity for benchmarking	Provide capacity for running/testing AI algorithm (GPU/CPU/RAM)
Virtual machine	Hosting node model, benchmarking and possibly VillasNode (therefore Linux)

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	*Requirement description*
RTDS data	Real time data exchange
None?	

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	*Requirement description*
TSO grid data	Only shared with Dutch node with NDA, needs to be synthetized (non-reverse-engineerable) to be shared with other parties
AI tool	Can only be shared after training with weights, no source code related to the training process.

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	*Task: Description*	*Objective*

5.3.3 German Node

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF (Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than TEF Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a TEF Test Use Case. These TEF Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The TEF Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	*Application(s)*	*Name of Use Case*
UC.X		Distribution Network Congestion Management for Renewable Integration

1.2 Version management

Version Management			
Version No.	*Date*	*Name of Author(s)*	*Changes*
0.1	20.01.2025	Ahmed Abdelgawad	Initial inputs
0.1	24.06.2025	Ana Romero Garcia	Additional comments
1.0	17.09.2025	Ahmed Abdelgawad	Finalizing template description

1.3 Business Use Case

Scope Description of the Business Use Case

Description	This describes the Business Use Case; the application of AI to a process, a challenge or issue in the energy sector. The aim of this use case is to manage congestion on distribution networks with a focus on managing powerflows and voltage on networks with higher penetrations of distributed energy resources that are renewable or higher loads. The challenges associated with this include variability and unpredictability in power generation, which can lead to issues with maintaining consistent power flow and voltage levels across the network. Managing these challenges effectively is crucial for ensuring stability and reliability in the power grid. Besides, a check on bird activity in the area is consider for the power flow management. AI can be used to analyze the data from distribution networks to predict and manage congestion. By assessing power flows and voltage levels, AI can identify potential overload situations.

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning Testing AI models for congestion management on distribution networks with renewable energy resources or higher loads.
Objective(s)	The use case intention; what is to be accomplished; who/what would benefit. - Establish a distribution system environment by simulation, without the limitations and restrictions imposed by the requirement for reliability in real systems for testing AI algorithms. - Focusing on a particular part of a network, establish the testing requirements for an AI system for the management of congestion on distribution networks to facilitate distributed energy resources. - Test AI algorithms and models on the modelled system and measure performance

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview. 150 words max The purpose of the TEF is to enhance AI algorithms aiming to support decision-making for distribution system operators. Therefore, the TEF provides an environment of a distribution system through simulation and emulation. This setup allows for the rigorous testing and validation of AI solutions, particularly in managing congestion and integrating distributed energy resources (DERs). By utilizing real operational data and advanced simulation tools like OpenDSS, the TEF enables researchers to assess the effectiveness of AI algorithms against traditional methods.
Complete description
Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. At the TEF, a simulator will be used to simulate a realistic model (electric and environmental) of the distribution network. Operational data is inputted, and various scenarios are designed to replicate potential congestion situations. The goal is to determine how AI can predict and manage these issues effectively. Once the simulation environment is ready, tests are run using AI algorithms designed for congestion management. The AI systems analyze the data coming from the simulator, predict congestion, and propose actions to mitigate it (taking into account those natural areas with high bird activity). Performance is monitored against traditional methods to assess whether AI can improve efficiency and response times.
Stakeholders

Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. • Distribution System Operators (DSOs) • Transmission System Operator (TSOs) • Energy suppliers
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. AI may mismanage congestion, leading to power outages or inefficiencies. AI system failures could result in unexpected congestion or system instability.
System's threats and vulnerabilities
Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. **Cyberattacks**: Targeting AI systems to manipulate data or disrupt operations. **Algorithmic Bias**: Leading to unfair treatment of certain segments or areas.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	*TEF Service*	*Objective of the service applied to the TEF Test Use Case*
1	AI-Tool Benchmarking	Test the developed AI agent on unseen data and benchmark against classical methods.
2	Data Provision	Facilitate the data transfer within the node between the node and stakeholders e.g. AI vendor.
3	Explainability Analysis	Provide feature importance analysis and contribution of each feature for AI decisions.
4	Mathematical Verification	Verify the developed AI agent

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
Renewable Energy Forecasting	Develop AI models to accurately predict renewable energy generation from sources such as solar, improving integration into the grid
Grid Resilience Assessment	Assess the resilience of the distribution network against various stress scenarios
Load Forecasting	Develop AI models to accurately predict the aggregated loads in the network from historic data.

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*
Data Synthetization	Grid data generation and anonymization for distribution network

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.

Please fill in a copy of the table for each TEF Service.

TEF Service: Data Provision		
	Name	***Description***
KPIs TEF Service implementation	Data Transfer Efficiency	Evaluates the speed and reliability of data transfer between simulator and AI agent.
	Latency	Assesses the time delay in data transmission from source to destination
KPIs TEF Service & Business Use Case combination	Easy Integration	Evaluates the success of data provision service in integrating with AI vendor systems.

TEF Service: AI Benchmarking		
	Name	***Description***
KPIs TEF Service implementation	Accuracy	Measures how accurately the AI tool's performance is assessed against established methodologies.
	Resource Utilization	Measures the computational resources used during benchmarking
KPIs TEF Service & Business Use Case combination	Maximum Line Loading under normal conditions (n-0)	Highest observed line loading under the base case (n-0), e.g., without any outages or contingency scenarios. This represents the peak stress on the grid in normal operation.
	Maximum Transformer Loading under normal conditions (n-0)	Highest observed transformer loading under base case (n-0). Evaluates transformer stress and helps identify equipment operating near or above thermal or stability limits under base case scenarios.
	Maximum Line Utilization under Contingency (n-1)	Maximum line loading across all single-contingency (n-1) scenarios. Indicates the worst-case stress under failure events of a single element.
	Maximum Transformer Loading under Contingency (n-1)	Maximum transformer loading encountered during any single-contingency (n-1) scenario. Assesses

		transformer performance under failure scenarios, revealing potential vulnerabilities in system design or operation.
	Longest Consecutive Overload Duration	Maximum number of consecutive time-steps a single grid element is overloaded.
	Setpoint Adjustments	Total magnitude of control action taken via generator, load or transformer tap setpoint changes (in MW, % or steps), categorized into increases and reductions. Quantifies the control effort required to maintain grid balance or mitigate violations.
	Setpoints Out of Range	Number or percentage of instances where generator or load setpoints exceeded their specified operational limits (e.g., Pmin/Pmax, Qmin/Qmax). Highlights control violations and potential risks due to ineffective setpoint management or insufficient operational margins.
	Bus Voltages Within Limits	Assessment of whether bus voltage magnitudes and angles remain within acceptable operational bounds across all buses. Magnitude violations may indicate risks like undervoltage collapse or equipment damage, and angle violations can signify potential oscillatory instability, or ineffective power transfer.

TEF Service: Explainability Analysis		
	Name	***Description***
KPIs TEF Service implementation	Feature Importance	Feature importance table is provided explains the contribution of each feature towards the prediction/forecasting
	User Accessibility	Assesses how easily users can access and interpret the explanations provided by the service
KPIs TEF Service & Business Use Case combination	Shapley Values	Explains the contribution of features as well as their combination in detecting congestion

TEF Service: Mathematical Verification		
	Name	***Description***
KPIs TEF Service implementation	Verification Results	Verification results are shared with proper explanation of the results
KPIs TEF Service & Business Use Case combination	Constraints Violation	Verifies that the agent is not violating any of the set constraints while operating on the grid.
	Verification	The entire pipeline of the AI algorithms verification is shared and made available for further usability

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate for the functionality of the TEF. Example: End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints). 3) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign. The TEF provides a simulated grid environment for a distribution network, which can be used for testing AI agents in the context of congestion management and maintaining stable levels of power flows and voltages.
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the TEF Service implementation KPI from above. AI-based models consistently demonstrate superior performance over classical optimization methods in managing congestion, effectively balancing power flows, maintaining stable voltage levels and grid loads.
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the TEF Service implementation KPI from above. AI-based models fail to outperform classical methods, showing minimal or no improvement in managing network congestion.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Example: Data privacy/security (data synthetization) Text	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Run the data synthetization on historical data and show that original data cannot be reverse-engineered. Text

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others. Optimization, Reinforcement Learning, Recommendation, sequential decision making/support
Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. Grid2op, OpenDSS

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/ https://www.etsi.org/committee/1640-sai
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. None

2.6 Challenges and issues

Challenges	***Mitigation***
Descriptions of challenges and issues of the TEF Test Use Case.	Explain how these challenges are addressed or mitigated.

Sharing of AI tool source code	Training is performed locally by the AI vendor, while benchmarking is conducted remotely on the node. To enable benchmarking without requiring to share the full source code, a challenge-based setup can be used—similar to formats seen in competitions like L2RPN or Flatland. In this setup, the benchmark provides a predefined entry point that needs to be implemented by the AI vendor and enables the system to query the agent for an action based on a given power grid observation. This allows the vendor to submit only the trained model (e.g., weights) along with the interface implementation. For local training, it is essential that the simulation environment is made accessible—either through a downloadable package or a separately hosted interface—for the AI vendor.

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. Trustworthiness assessment: Assess how trustworthy is the AI agent and how does it perform under uncertainties
Scalability	Discuss how the use case can be scaled for future needs.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case
Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SGD that are within the scope of this use case.

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
		report, mandates and regulatory constraints, paper, patent, press release	Public / confidential	Where does the document influence the use case?		

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[1]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

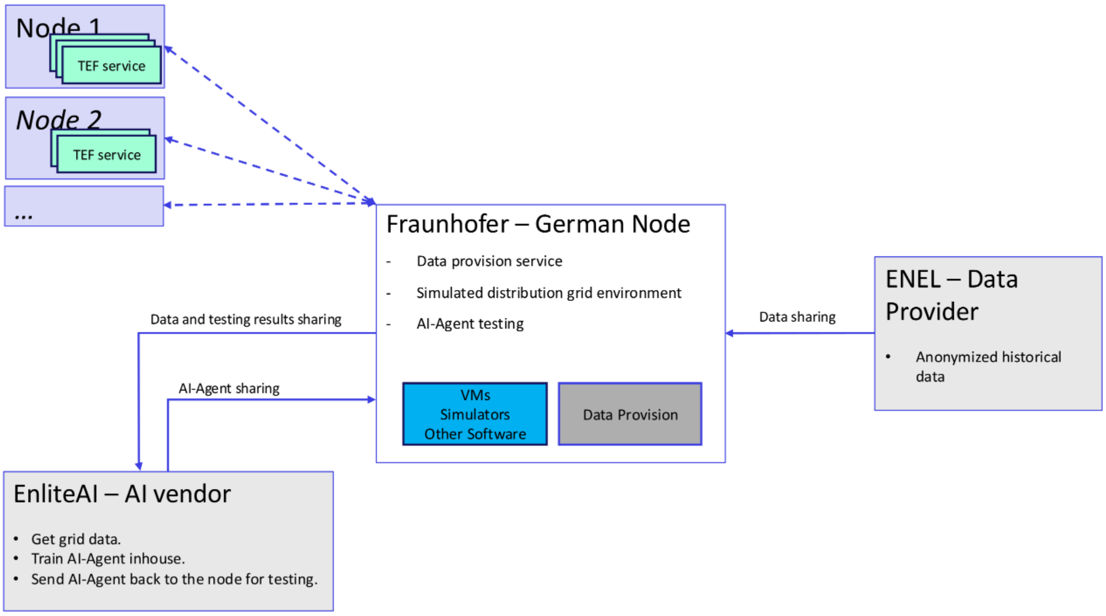
Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Involved Partners	*Tasks*	*Requirements for Data Exchange*
ENEL	Data provider	Data can be exchanged directly. It will be anonymized data (without references to the grid.)
EEU	Grid simulation tool	
Fraunhofer	Node infrastructure	
RWTH	VILLASnode	
ENLITE	Trains its AI tool with provided data, tests it against the benchmark and refines the tool.	Sharing the AI tool source code not intended. Trained AI tool can be shared (e.g., weights).

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	*Description of the step*	*Time constraint*	*Data format*
Enel historical grid data	– ENEL will provide data grids and the measure from a data base (historical real data). Data grids will be anonymized (without references) –	Each 15 min	JSON / CSV files
ENLITE congestio n tool using Grid2op	Grid2Op serves as middleware between the AI tool and the power system simulator (e.g., OpenDSS, PandaPower). At each time step, it collects injection data—comprising generation and load—and forwards it to the simulator. In return, it retrieves the resulting grid state and presents it to the AI tool. Based on this information, the AI selects from a predefined set of actions, such as topology reconfiguration, generator setpoint adjustments, or transformer tap changes and many more, which Grid2Op then applies to the simulation.	Although the grid data is provided in 15-minute resolution, the simulation and AI decision-making can operate at a much finer time granularity.	
OpenDSS	Applies setpoints and feeds congestion tool to receive new setpoints		Reads and writes.dat files



3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	*Use*
Panda Power	Grid Data
	OpenDSS
	ENLITE AI Tool

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

	Requirement description
Programming language	Python , C++
Input (data format)	CSV, JSON
Output (data format)	CVS, JSON
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing is not intended; it should only run on-premises at the AI vendor.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning)

	Reinforcement Learning, optimization (depends on method), sequential decision making
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) The AI tool aims to perform congestion management in an electrical distribution grid. It monitors the grid state and decides on corrective actions (such as redispatching generation, curtailing loads, or reconfiguring the network topology) to avoid voltage overloads, maintain system stability, and ensure operational security.
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Neural Network and Tree-based model, optimization (depends on method)
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Graph-based models, tabular models, optimization models (depends on method)
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Maze-rl, Pytorch, Scikit-learn, Tensorflow, The specific packages used in the implementation are yet to be defined.

Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints?
	The problem is a sequential decision-making and control problem with a strong optimization component. The AI must decide corrective actions (such as redispatch, curtailment, or topology changes) to minimize or eliminate congestion on distribution lines while satisfying operational constraints.
	
	• Objective: Keep all line flows within their thermal limits (no overloads); no violation of operational constraints (voltage limits).
	• Constraints:
	    o Physical constraints: Power flow equations.
	    o Operational constraints: Generator limits, load balancing, topology restrictions.
	• Variables:
	    o Discrete: Topology actions (e.g., connecting lines to different buses).
	    o Continuous: Redispatching generator outputs or curtailing load.
	• Constraint types:
	    o Non-linear constraints: AC power-flow.
	    o Mixed-integer nature: Both continuous (redispatch) and discrete (topology switching) variables.

Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? The verification goal is to ensure that all operational constraints are respected across the scenarios handled by the AI agent. The focus is on: • Constraint satisfaction: Verifying that the actions proposed by the agent avoid violations of voltage limits on distribution lines (no overvoltage). • Robustness testing: Checking the agent's behavior under varying grid conditions (e.g., different contingencies, fluctuating loads, and renewable generation).

	• (Distance to optimal solution: Optionally measuring how close the AI agent's performance is to the best-known (optimal) solution in terms of system disruptions.)

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	*Requirement description*
Runtime	Runtime of the AI algorithm should not exceed X days
Computational cost	The AI tool should be capable of running efficiently on a virtual machine with moderately specifications. As a guideline: CPU: ~32 cores GPU: NVIDIA RTX 4090 (or equivalent) RAM: ~250 GiB These specifications may be adjusted or upgraded in the future.

3.6 Requirements for Data Synthetization

Data synthetization may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data that may not be shared by the stakeholder. Which requirements are needed for the synthetization?

Data	*Data description*	*Real data/historical data/synthetic data*	*Requirements on Privacy*	*Description of case for synthesizing data*
Enel Grid data	*anonymized*	*Real data /Historical data*	NO (Anonymized)	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	*Requirement description*
Data manipulation	Offset calculation, alignment of timestamps, …

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	*Requirement description*
Virtual machine	Hosting node model, simulation and possibly VILLASnode (therefore Linux)

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	*Requirement description*
Grid Data Exchange	Data provided by Enel has 15 min resolution, but data exchange between AI agent and simulator might be faster

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	*Requirement description*
AI Tool	Can only be shared after training with weights, no source code related to the training process.
Data	Data is not intended to be shared with other entities rather than the AI vendor

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	*Task: Description*	*Objective*
	Data set: Get different data	Test AI algorithm with new data
	Real-time simulator: calculated setpoints by AI algorithm on a simulated grid	Verify that AI algorithm can cope with dynamic inputs/outputs

5.3.4 Portuguese Node

Performance evaluation of AI-powered Virtual Energy Manager

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each

	node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	*Application(s)*	*Name of Use Case*
UC.X		Performance evaluation of AI-powered Virtual Energy Manager

1.2 Version management

Version Management			
Version No.	*Date*	*Name of Author(s)*	*Changes*
0.1	14.05.2025	Miguel Carvalho	First draft

1.3 Business Use Case(s)

Priority	*Business Use Case*	*Scope Description of the Business Use Case*
1	Name Business Use Case	This describes the Business Use Case; the application of AI to a process, a challenge or issue in the energy sector. Example: TSO operational planning AI-powered Virtual Energy manager tool (WIS4Households) aimed towards residential users that supports them to become more energy efficient by providing targeted and quantified energy efficiency measures generated via a combination of different Data Analytics models, including Non-Intrusive Load Monitoring (NILM) algorithms

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning Validating and improving the performance of AI-based tools to promote energy efficiency
Objective(s)	The use case intention; what is to be accomplished; who/what would benefit. (1) Promote energy efficiency at the residential level via an AI-based tool that will use DSO collected smart metering data to provide targeted recommendations to its users. (2) Identify and quantify potential energy savings from energy efficiency measures adoption. (3) Provide higher knowledge regarding energy consumption at the household level through the use of AI-based tools such as Non-Intrusive Load Monitoring algorithms. (4) Provide targeted and quantified energy efficiency recommendations to its users.

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview. 150 words max The TEF relates mainly to making available an AI-powered tool (WIS4Households) to residential users that supports them to become more energy efficient by providing targeted and quantified energy efficiency measures generated via a combination of different Data Analytics models. WIS4Households uses DSO collected smart metering data and applies an AI-based Non-Intrusive Load Monitoring (NILM) algorithm to estimate the energy used in each main electrical appliance. Based on the outputs generated by the NILM the most adequate energy efficiency measures and generated and quantified in terms of investment needed and savings generated. This TEF will make possible to evaluate and improve the impact that AI-based tools can have in the promotion of energy efficiency and CO_2 emissions reductions while generating tangible results for its user via energy bill reduction.
Complete description

Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. NILM algorithms and energy efficiency recommendation modules are of critical importance to Home Energy Management Systems (HEMS) since it allows these AI-based tools to present users new layers of information, such as how much they are spending on their appliances and the most adequate energy efficiency measures to be adopted, rather than just a simple load consumption and/or production profile. These new layers of information effectively can be delivered relying only on DSO collected smart metering data (without the need of additional CAPEX on higher resolution meters or smart plugs), thus being a low-cost and fully scalable solution. The expected use of the TEF is the following: (1) The DSO makes available energy consumption data from their clients to be ingested by the WIS4Households solution; (2) The user accesses WIS4Households; (3) Periodically, a set of WIS4Households back-end jobs, run the NILM module to estimate the energy consumption for each load disaggregation class provided as output. (4) The NILM output is then used as input for the energy efficiency measures generator to provide targeted and quantified energy efficiency measures for each WIS4Households user. (5) The user can input additional domain-specific data (such as appliance related metadata) to support NILM outputs and targeted energy efficiency measures to become more accurate and relevant;
Stakeholders
Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. Energy suppliers; 3rd parties interested in provided energy efficiency related tools; network operators; customers/end-users (e.g., residential electricity consumers)
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. Being a fully algorithmic AI-powered module, the NILM outputs are subject with Trustworthiness issues related to the accuracy it can achieve (that can be measured with different indicators such as event recognition %, number of false positives or negatives, energy usage load disaggregation coverage, etc.).
System's threats and vulnerabilities
Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. AI-based NILM modules that are not robustly evaluated and tested, or that have a low performance, can lead to sub-optimal or incorrect decision-making due to poor predictive information provided to its users. This directly affects, for instance the capability to recommend the most adequate energy efficiency measures to each user, thus limiting its impact potential in improving energy efficiency of its users.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data Provision	Obtain DSO smart meter readings with 15 minute time interval.
11	AI Tool Benchmarking	Performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***
Validation of Load and RES forecasting algorithms	

Performance evaluation of energy-sharing mechanism	
Testing of DER scheduling/control algorithms	
Performance evaluation of network monitoring algorithm	

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
AI-powered virtual energy management	Improve energy efficiency in households based on smart metering data with load disaggregation and tailored recommendations
Optimization of energy community operation	Management of energy communities including AI-powered features such as optimal scheduling of community storage and optimal sharing coefficients for participants
Voltage Control with distributed energy resources	Algorithms for voltage control of flexible distributed energy resources based on AI models for state estimation
Edge load control for flexibility services	Provide flexibility services with control of flexible loads (Storage, EWH, Heat pumps) based on edge AI and cloud solutions
Network topology estimation	AI-based models to infer the topology of the electrical grid and fix misspecified grid configurations (e.g., consumer phases)
State estimation and network quality	AI-based state estimation algorithms to improve observability and control of the electrical grid, including network quality metrics

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Data Provision		
	Name	*Description*
KPIs TEF Service implementation	Data access	The AI vendor can receive data from the TEF to train the AI model

KPIs TEF Service & Business Use Case	Data availability	Historical metering data from the DSO is enough to train the AI model

TEF Service: AI Tool Benchmarking		
	Name	**Description**
KPIs TEF Service implem	Benchmark comparison	A baseline/state-of-the-art AI tool or ranking list is given to compare the results of the benchmarked AI tool and rank its performance.
KPIs TEF Service &	Equipments consumption identified by NILM (Non-Intrusive Load Monitoring)	Compares the consumption for each equipment identified by the AI-model in simulation or lab-based testing when compared to other models.

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate for the functionality of the TEF. Example: End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints). 3) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign. The application of AI-based tools on top of DSO collected smart metering data can lead to improved energy efficiency levels on the residential sector.
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the TEF Service implementation KPI from above. The availability of AI-powered tools such as WIS4Households provides new layers of information towards their users such as Load Disaggregation and targeted and quantified energy efficiency recommendations that ultimately contribute to the achievement of higher levels of energy efficiency.
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the TEF Service implementation KPI from above. It was not possible to validate that the application of AI-powered modules and the new layers of information made possible such as Load Disaggregation and targeted and quantified energy efficiency recommendations, resulted on the improvement of energy efficiency levels on residential users.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Example: Data privacy/security (data synthetization) Operational Data Provision	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Run the data synthetization on historical data and show that original data cannot be reverse-engineered. Acquisition of smart metering data from the DSO with a 15-minute time resolution on a daily basis for all end-users.

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others. Recognition and recommendation

Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. None

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/ https://www.etsi.org/committee/1640-sai Text
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. Text

2.6 Challenges and issues

Challenges	Mitigation
Empowering residential users to make more informed decisions towards energy efficiency	In WIS4Households we seek to provide additional layers of reasoning, such as load disaggregation and targeted and quantified energy efficiency measures, that go well beyond the typical energy consumption and production profiles, that are commonly present in typical HEMS.
To be massively adopted the solution needs to be highly Scalable	Develop low-cost and scalable solutions based on already existent smart metering data.

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. Include forecasting tools for consumption, production, weather and market spot price, so that in the case of users with market indexed tariffs, WIS4Households can recommend optimal energy appliance usage for the next days.
Scalability	Discuss how the use case can be scaled for future needs. WIS4Households is an AI-powered Virtual Energy Manager that relies solely on DSO collected smart metering data, without requiring any additional CAPEX in higher resolution smart meters or smart plugs. This characteristic makes it a highly scalable AI-powered solution that can be applied in any energy grid that has an AMI infrastructure.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case Integrating external data may increase the risk of exposing sensitive or proprietary information; Assure that data provision and synthetization do not lead to privacy breaches.
Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SGD that are within the scope of this use case. 7, 11

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
		report, mandates and regulatory constraints, paper, patent, press release	Public / confidential	Where does the document influence the use case?		

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[3]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	***Tasks***	***Requirements for Data Exchange***
Electricity retailer	Data provider	Data cannot be exchanged directly because of security
R&D Institute	Node host, Data provider, AI development	None
Technology company	AI development	

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	***Description of the step***
Source	Smart metering data is collected from residential users and ingested into the WIS4Households platform.
Time resolution	The input data from the database is sampled at a resolution of 15 minute interval.
Destination	The end-user accesses the platform to view appliance usage and personalized energy-saving recommendations.

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	***Use***

[3] VILLASnode | VILLASframework

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

	Requirement description
Programming language	Python
Input (data format)	Smart meter data with 15 min interval and metadata from the end-users
Output (data format)	Disaggregation and energy eficiency measures
Interface	Protocol? API? User engagement web app
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing not intended this should only run locally in node.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Supervised learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Regression (NILM) + Recommendation generation (energy efficient measures)
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Logistic Regression
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series models (NILM) and tabular models (energy efficient measures)
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Scikit-learn
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? The core problem is energy disaggregation, treated as a regression problem. Constraints: - Only continuous variables (kWh) - Time-aligned consumption data - User-specific constraints like appliance presence.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? - Compare recommendations with actual savings. - Test robustness to noisy inputs or missing timestamps.

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***
Robustness testing	The model must handle missing values, timestamp misalignments, and consumption anomalies without failure.
Impact of energy efficient measures	Recommendations will be cross-checked against historical or simulated user feedback to assess accuracy in projected savings.

3.6 Requirements for Data Synthetization

Usage of the TEF service “Data Synthetization” [4] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

	Requirement description
Need for synthetization	Data privacy/security, increased variety of data for AI tool training/testing? Text
Type of data	Grid connectivity, substation configuration, circuit parameters or injection data? Text
Time details of data	Time series or snapshot data, minute/hourly/daily/… resolution, time horizon of one day/month/year/… Text
Location of data synthetization	Where should data synthetization take place? Could the data be sent to the Dutch node to synthetize it there or not? Text
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	***Requirement description***
Timestamp alignment	Consumption data must be aligned to regular intervals (e.g., every 15 minutes). Missing timestamps must be interpolated or handled gracefully.
User ID anonymization	Data must be pseudonymized before processing to ensure user privacy and GDPR compliance.
Appliance metadata mapping	Optional user-provided metadata (e.g., appliance ownership) needs to be linked correctly with NILM output for accurate recommendations.
Input format normalization	Smart meter data from DSO must be converted to a unified format (JSON with fields: timestamp, consumption).

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	***Requirement description***

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	***Requirement description***

[4] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	***Task: Description***	***Objective***

Performance evaluation of energy-sharing mechanism

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	***Application(s)***	***Name of Use Case***

UC.X		Performance evaluation of energy-sharing mechanism

1.2 Version management

Version Management			
Version No.	***Date***	***Name of Author(s)***	***Changes***
0.1	14.05.2025	Luís Moncaixa	First draft

1.3 Business Use Case(s)

Priority	***Business Use Case***	***Scope Description of the Business Use Case***
1	Name Business Use Case	This describes the Business Use Case; the application of AI to a process, a challenge or issue in the energy sector. Example: TSO operational planning Optimization of energy-sharing coefficients in an energy community based on AI-driven models for forecasting consumption, production, and available surplus, along with optimized scheduling of flexible assets such as storage systems.

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	The scope defines the limits of the TEF Test Use Case. Example: validating an algorithm for network planning Validating the impact of optimized energy-sharing keys on the overall benefits for the community and its participants.
Objective(s)	The use case intention; what is to be accomplished; who/what would benefit. 1) Maximize benefits for all community participants by increasing their savings. 2) Increase the volume of shared energy within the community while ensuring fairness among participants and minimizing surplus. 3) Enable more efficient management of flexible assets, such as storage systems and electric vehicles. 4) Energy suppliers also benefit, as the increased volume of shared energy within the community reduces their energy procurement costs.

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Short text intended to summarize the main idea of using the TEF. This description should help the reader searching for a use case or looking for an overview. 150 words max This TEF explores the application of AI-driven models to optimize energy-sharing coefficients within an energy community. By improving forecasting of consumption, production, and available surplus, the system enhances energy distribution, ensuring fairness and maximizing benefits for all participants. Optimized sharing leads to increased individual and collective savings, reduced energy surplus, and more efficient management of flexible assets such as storage systems and electric vehicles. Additionally, energy suppliers benefit from lower procurement costs due to a higher volume of shared energy. The TEF will be used to validate the impact of these optimizations, assessing their effectiveness in improving economic, operational, and environmental outcomes.
Complete description
Provides a complete narrative of the TEF Test Use Case from a user's point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so non-domain experts can understand it, and can (should) have a step-by-step description. The complete description of the TEF Test Use Case can range from a few sentences to a few pages. This section often helps the domain expert think through the function's user requirements before getting into the details required by the next sections of the TEF Test Use Case. In an energy community, participants generate, consume, and share energy between them. The challenge is to fairly distribute surplus energy while maximizing economic and operational benefits. This TEF tests the application of AI-driven models to optimize energy-sharing coefficients, ensuring efficient and equitable distribution. The expected use of the TEF is the following:

1) Based on the consumption, generation and surplus forecasts, the system calculates the optimal energy-sharing coefficients for each participant by each producer in the community. 2) The main goal is to maximize individual and collective savings while reducing surplus energy. 3) The optimized coefficients are applied to allocate energy fairly among participants. 4) The TEF evaluates economic benefits and operational impacts in comparison with the sharing mechanisms normally used under current regulations (fixed and proportional coefficients). By using AI to optimize energy-sharing, participants save more, excess energy is minimized, and the entire community benefits from improved efficiency and sustainability.
Stakeholders
Stakeholders that can affect or be affected by the AI system in the scenario, e.g., organizations, customers, third parties, end-users, the community, the environment, negative influencers, bad actors, etc. Community manager; electricity suppliers; customers/end-users (e.g., energy community participants)
Stakeholders' assets, values
Stakeholders' assets and values that are at stake with potential risk of being compromised by the AI system deployment — e.g., competitiveness, reputation, trustworthiness, fair treatment, safety, privacy, stability, etc. Fair treatment, privacy, stability
System's threats and vulnerabilities
Threats and vulnerabilities can compromise the assets and values mentioned above – e.g., different sources of bias, incorrect AI system use, new security threats, challenges to accountability, new privacy threats (hidden patterns), etc. Low-performance forecast data will impact the optimization of the best sharing coefficients for each participant, thereby minimizing the benefits obtained, such as the savings generated from the received energy. Additionally, it will affect energy management within the community, as fewer energy transactions may occur between participants. Data privacy of consumers is also a threat that must be accounted for.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data Provision	Obtain DSO smart meter readings with 15 minute time interval.
8	AI Tool Benchmarking	Performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***
Performance evaluation of recommendation algorithm for energy efficiency	
Validation of Load and RES forecasting algorithms	
Testing of DER scheduling/control algorithms	
Performance evaluation of network monitoring algorithm	

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***

AI-powered virtual energy management	Improve energy efficiency in households based on smart metering data with load disaggregation and tailored recommendations
Optimization of energy community operation	Management of energy communities including AI-powered features such as optimal scheduling of community storage and optimal sharing coefficients for participants
Voltage Control with distributed energy resources	Algorithms for voltage control of flexible distributed energy resources based on AI models for state estimation
Edge load control for flexibility services	Provide flexibility services with control of flexible loads (Storage, EWH, Heat pumps) based on edge AI and cloud solutions
Network topology estimation	AI-based models to infer the topology of the electrical grid and fix misspecified grid configurations (e.g., consumer phases)
State estimation and network quality	AI-based state estimation algorithms to improve observability and control of the electrical grid, including network quality metrics

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service:Data Provision		
	Name	*Description*
KPIs TEF Service implementation	Data access	The AI vendor can receive data from the TEF to train the AI model
	Data privacy	Ensure anonymization and secure processing of user data.
KPIs TEF Service & Business Use Case	Data availability	Historical metering data from the DSO is enough to train the AI model

TEF Service: AI Tool Benchmarking		
	Name	*Description*
KPIs TEF Serv ice	Benchmark comparison	A baseline/state-of-the-art AI tool or ranking list is given to compare the results of the benchmarked AI tool and rank its performance.

	Data flow	The service is accessible by the AI vendor and the outputs can be shared.
KPIs TEF Service & Business Use Case	Improvement on Financial Savings per Participant	% increase in savings for each community participant in simulation or lab-based testing when compared to other models.
	Increase in Shared Energy Volume (kWh per month)	Compares the amount of shared energy in simulation or lab-based testing when compared to other models.

2.2 Hypotheses of TEF Test Use Case

Hypothesis	State the hypothesis that you are having and wish to evaluate for the functionality of the TEF. Example: End-to-end verification. This requires alignment between test data, mathematical models, and lab components. The goal is to create a process: 1) AI tools are trained on a set of data (test data may or may not be part of the data the AI tool has trained on). 2) Mathematical Verification tests if the AI tool violates any constraints across the whole operating region (continuous range of datapoints). 3) The lab tests assess the performance of the tool in an operationally relevant environment through a well-designed experiment campaign. The application of optimized sharing coefficients within the community will increase the overall benefits for all participants, leading to higher savings while reducing the amount of surplus energy through increased energy sharing among participants, always ensuring fairness.
Success (scenario 1)	Specify what defines the hypothesis as successful referring to the TEF Service implementation KPI from above. The application of optimized sharing coefficients leads to measurable benefits for the energy community. Specifically, participants should experience increased financial savings, while the amount of surplus energy decreases due to improved sharing efficiency. The system must also ensure fair energy distribution, preventing significant disparities among participants. Additionally, a higher percentage of locally generated energy should be consumed within the community, reducing reliance on external sources. If these conditions are met, the TEF testing will confirm that AI-driven optimization effectively improves economic, operational, and environmental outcomes in energy communities.
Unsuccessful (scenario 2)	Specify what defines the hypothesis as unsuccessful referring to the TEF Service implementation KPI from above. If the application of optimized sharing coefficients fails to generate measurable benefits for the energy community. This includes scenarios where participants do not experience increased financial savings, or where surplus energy remains high due to inefficient sharing. If the system leads to unfair energy distribution, creating significant imbalances among participants, it would indicate failure. If these conditions occur, the TEF testing would demonstrate that AI-driven optimization does not significantly enhance energy community performance.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Example: Data privacy/security (data synthetization) Operational Data Provision	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Run the data synthetization on historical data and show that original data cannot be reverse-engineered. Acquisition of smart metering data from the DSO with a 15-minute time resolution on a daily basis for all end-users.
Data Synthesization	Run the forecast models for consumption, production, and surplus, and use these data as input for the optimization model of energy-sharing coefficients

2.4 Features of TEF Test Use Case

Method(s)	AI method(s)/framework(s) used by AI tool provider. A pull-down list includes the following terms: recognition, natural language processing, knowledge processing and discovery, inference, planning, prediction, optimization, interactivity, recommendation and others. Optimization

Platform	Indicate here the digital environment: Grid2Op, Flatland, BueSky. None

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/ https://www.etsi.org/committee/1640-sai Text
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. Text

2.6 Challenges and issues

Challenges	***Mitigation***
Ensuring Fairness in Energy Distribution	Develop transparent allocation rules that consider consumption, generation capacity and surplus energy available. Provide clear communication and allow adjustments based on participant feedback.
Scalability and Interoperability	Design optimization models that can adapt to different community sizes and grid conditions. Ensure compatibility with various energy management systems and standards.

2.7 Future Scope

Future Scope	
Enhancements	Suggest potential enhancements or extensions to the use case beyond the scope of the AI-EFFECT. One option could also be to include other TEF Services. Calculate the optimal share coefficients in a near real-time environment without need the forecast data. Additional TEF services can be considered.
Scalability	Discuss how the use case can be scaled for future needs. The model can be scaled to larger communities, including entire neighborhoods, commercial districts, or even regional grids. With proper integration, AI-based optimization can manage energy distribution across thousands of participants, improving overall system efficiency.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Description of societal concerns related to the use case Integrating external data may increase the risk of exposing sensitive or proprietary information; Assure that data provision and synthetization do not lead to privacy breaches.
Sustainable Development Goals (SGD) to be achieved
The Sustainable Development Goals (SDGs), https://sdgs.un.org/goals, are a collection of 17 global goals set by the United Nations General Assembly. SDGs are a universal call to action to end poverty, protect the planet and ensure that all people enjoy peace and prosperity. Indicate here the SGD that are within the scope of this use case. 7, 11

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***
		report, mandates and regulatory constraints, paper, patent, press release	Public / confidential	Where does the document influence the use case?		

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[5]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	***Tasks***	***Requirements for Data Exchange***
Electricity retailer	Data provider	Data cannot be exchanged directly because of security
R&D Institute	Node host, Data provider, AI development	None
Technology company	AI development	

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	***Description of the step***
Source	Historic data is provided by DSO, including consumption and surplus records from an energy community. This historic data can also be provided by the AI forecast models.
Time resolution	The input data from the database is sampled at a resolution of 15 minute interval.
Destination	Watt-IS receives the pre-processed data to realize the optimization model. Once the optimization is complete the resulting coefficients will be shared with the DSO to be used on a real-time environment that mimics the operational behavior of an energy community, including flexible assets like batteries.

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	***Use***

[5] VILLASnode | VILLASframework

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

	Requirement description
Programming language	Python
Input (data format)	JSON containing time-series data on consumption and surplus for each community participant
Output (data format)	JSON structure with time-series optimized energy-sharing coefficients by producer-consumer combination
Interface	Protocol? API? RESTful API
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Sharing not intended this should only run locally in node.
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Optimization algorithm, statistical learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) optimize energy-sharing coefficients to maximize savings and minimize surplus
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods linear programming
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series models and tabular data
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) Text
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? This is an optimization problem with linear constraints. Objective: maximize community-wide savings while ensuring fair distribution.
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? We verify solution feasibility (no energy over-allocation), compare performance with traditional static sharing (baseline), and test stability under synthetic variations. Robustness testing includes sensitivity analysis to input deviations.

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***
Runtime	Runtime of the AI algorithm should not exceed 2 minutes for a 24-hour forecast period, assuming a community of up to 100 participants.
Optimization success rate	Optimization algorithm must converge to a valid solution in ≥95% of test cases.

3.6 Requirements for Data Synthetization

Usage of the TEF service "Data Synthetization" [6] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

Requirement description	
Need for synthetization	Data privacy/security, increased variety of data for AI tool training/testing? Text
Type of data	Grid connectivity, substation configuration, circuit parameters or injection data? Text
Time details of data	Time series or snapshot data, minute/hourly/daily/… resolution, time horizon of one day/month/year/… Text
Location of data synthetization	Where should data synthetization take place? Could the data be sent to the Dutch node to synthetize it there or not? Text
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	***Requirement description***
Data manipulation	Input data (consumption, surplus) requires timestamp alignment across different sources
Input format	Input data must be provided in JSON format, containing fields such as timestamp, participant_id, consumption, surplus.

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	***Requirement description***

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	***Requirement description***

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***

[6] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	*Task: Description*	*Objective*

Testing of DER scheduling/control algorithms

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	*Application(s)*	*Name of Use Case*
UC.2		Testing of DER scheduling/control algorithms

1.2 Version management

Version Management			
Version No.	*Date*	*Name of Author(s)*	*Changes*
0.1	25.02.2025	Carlos Silva	First draft
0.2	16.05.2025	Carlos Silva	Adapt to new template

1.3 Business Use Case(s)

Priority	Business Use Case	Scope Description of the Business Use Case
1	Voltage control with distributed energy resources	Algorithms for voltage control of flexible distributed energy resources based on AI models designed to solve/avoid technical violations
2	Edge load control for flexibility services	Provide flexibility services with control of flexible loads (Storage, EWH, Heat pumps) based on edge AI and cloud solutions

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	Testing AI-based algorithms for control of distributed energy resources
Objective(s)	(1) Development of an environment where TEF users and AI developers can enhance and rigorously test their control or scheduling algorithms for distributed energy resources, considering multiple potential objectives: a. Minimize power adjustments (i.e., flexibility) to avoid/solve voltage violations b. Define dynamic operating envelopes for each flexible resource that ensure compliance with technical constraints c. Maximize the use of renewables in scheduling algorithms (e.g., EV charging) without compromising user comfort d. Implement distributed approaches that leverage the computing power of edge devices (e.g., EV chargers) to reduce data transfer and/or protect private data (2) Ensure that users can access operational and external data by providing access to relevant data sources and tools. (3) Ensure that users can enhance their solutions, by providing access to a pipeline consisting of performance assessment or evaluation tools, baseline models for benchmarking, and mathematical verification algorithms. (4) Provide complex testing environments for AI-based algorithms, including simulated digital environments and laboratory-based experiments with real assets and distributed energy resources. (5) This use case is expected to benefit providers of AI solutions directly, by improving the development phase of their model, namely exposing them to simulated and laboratory-based environments that are close to real-world conditions. It is also expected that the use case indirectly improves other stakeholders that use these solutions for decision-making or as end-users (e.g., network operators' congestion management, or end-users' demand-side management).

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
Low-voltage (LV) networks play a key role in the new operational paradigm but remain a major bottleneck due to several challenges: (a) limited sensor coverage and lack of near real-time, reliable communication between smart meters and the distribution system operator (DSO), highlighting the need for data-driven state estimation methods with minimal data requirements; (b) low levels of automation, with fuse-based protection and manual breakers, despite the potential of storage systems and smart appliances to support the grid; and (c) increasing integration of renewables and storage at the consumer level, often combined with incomplete or outdated information on grid topology and electrical characteristics, which calls for data-driven methods to estimate topology, parameters, and behind-the-meter generation capacity. The absence of reliable or updated information about network assets limits the use of conventional analytical techniques for controlling and managing LV grids, such as optimal power flow (OPF) and analytical sensitivity factors, which rely on accurate grid topology and electrical parameters. As an alternative, data-driven methods can estimate sensitivity factors directly from smart meter measurements of active power and voltage, without requiring a detailed grid model. The use of the TEF relates mainly to its data provision, benchmarking, and testing capabilities in simulated and real environments. This use case not only improves and validates AI models that relate to control of distributed

energy resources in LV grids, but also creates value for AI solution developers by ensuring that their models are tested in different kinds of environments (simulation and lab-based experiments, utilizing both virtual and physical assets) to support and validate their deployment in real-world applications. More specifically, using the TEF allows this use case to indirectly enhance the resource optimization capabilities at different levels (households, energy communities, LV grids) and related key performance indicators, since resource scheduling and control depend on the algorithms used to compute preventive or real-time control actions.
Complete description
Control and scheduling algorithms for distributed energy resources are critical for applications in energy systems, since for instance the safe operation of the electrical grid can depend on their performance. Using a pipeline of TEF services can enhance the performance of the algorithms and make them more robust in real-world deployment through access to complex and thorough testing environments. The expected use of the TEF is the following: (1) The user deploys the DER control/scheduling algorithm in the TEF. (2) The user can request domain-specific data (operational and external) from the TEF, to support model training and complement the available proprietary dataset. Alternatively, the user can request synthetized data, ensuring data privacy concerns are met. (3) The user can request a performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks. (4) The user can request a mathematical verification, to identify critical operating regimes for the AI tool specifically for control or optimization purposes. (5) The user can request a simulation-based test, which will result in access to a simulated digital testing environment with the necessary tools to test the AI tool under diverse conditions close to real-world applications. (6) The user can request a lab-based test, which will result in access to a real facility (and related assets) within a lab-based framework, where the AI tool can be tested under real operating conditions.
Stakeholders
Network operators; electricity suppliers; customers/end-users (e.g., residential electricity consumers, energy community participants).
Stakeholders' assets, values
Safety of controllable devices, data privacy.
System's threats and vulnerabilities
Control and scheduling algorithms that perform poorly can lead to sub-optimal decision making. Consequently, sub-optimal control actions can lead to poor performance of distributed energy resources, to poor performance of energy-related KPIs in households (cost, self-consumption of renewable energy), or even to issues related to the LV grids (e.g., voltage violations).

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data provision	Domain-specific data (operational) to support model training and complement the available dataset.
4	Data Synthetization	Synthetic data to ensure data privacy concerns are met.
4	AI Tool Benchmarking	Performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.
4	Mathematical Verification	Identify critical operating regimes for the AI tool.
2	Virtual Facility Access	Provide access to virtual assets for simulation-based testing of control algorithms.
1	Simulation-based testing	Control algorithms are evaluated under a set of diverse operating conditions in a simulated digital environment.
2	Experimental design for lab-based testing	Suitable experimental design based on a specification of the testing need.
2	Facility Access	Provide access to physical assets for lab-based testing of network monitoring algorithm
2	Lab-based testing	Control algorithms are evaluated under a set of diverse operating conditions in a real-world environment with physical assets.
3	Living lab testing	Control algorithms are evaluated under a real-world operating conditions with real consumers and infrastructure

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
Performance evaluation of recommendation algorithm for energy efficiency	
Performance evaluation of energy-sharing mechanism	
Validation of forecasting algorithms	
Performance evaluation of network monitoring algorithm	

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
AI-powered virtual energy management	Improve energy efficiency in households based on smart metering data with load disaggregation and tailored recommendations.
Optimization of energy community operation	Management of energy communities including AI-powered features such as optimal scheduling of community storage and optimal sharing coefficients for participants.
Network topology estimation	AI-based models to infer the topology of the electrical grid and fix misspecified grid configurations (e.g., consumer phases).
State estimation and network quality	AI-based state estimation algorithms to improve observability and control of the electrical grid, including network quality metrics.
Forecasting electric vehicle charging patterns	AI-based models for prediction of EV charging sessions to support control of flexible resources.
Load and RES generation forecasting	AI-based models for load and RES generation forecasting.

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*
Conformity Assessment	Ensure the AI solution meets industry standards and legal requirements

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Data Provision		
	Name	***Description***
KPIs TEF Service	Data access	The AI vendor can receive data from the TEF to train the AI model

KPIs TEF Service & Business Use Case	Data access	Load and RES generation data from the DSO are provided to the AI vendor via TEF on demand to support voltage control
	Data availability	Historical metering data from the DSO is enough to train the AI model

TEF Service: Data Synthetization		
	Name	***Description***
KPIs TEF Service implementation	Data privacy	Original data cannot be reverse-engineered from synthetic data.
	Data access	The AI vendor can receive synthetized data from the TEF to train the AI model
	Operation modes	Synthesizer module can provide different types of data
KPIs TEF Service & Business Use Case	Synthetized Load and RES generation data access	The AI vendor can receive synthetized load and RES generation data from the DSO via TEF on demand
	Synthetic data quality metrics	Similarity between synthetized load and RES generation data and real data (e.g., time-series based properties).

TEF Service: AI Tool benchmarking		
	Name	***Description***
KPIs TEF Service implementation	Benchmark comparison	A baseline/state-of-the-art AI tool or ranking list is given to compare the results of the benchmarked AI tool and rank its performance.
	Data flow	The service is accessible by the AI vendor and necessary data can be shared. This includes the trained AI tool and benchmarking results.
KPIs TEF Service & Business Use Case combination	Renewable energy self-consumption improvement	% increase in renewable energy self-consumption in simulation or lab-based testing when compared to other models
	Reduction of voltage violations	% reduction in voltage violations in simulation or lab-based testing when compared to other models
	Reduction of severity of voltage violations	% reduction in severity of voltage violations in simulation or lab-based testing when compared to other models
	Ability to meet flexibility requests	% of the flexibility requests that were met with the scheduling of DER.
	Ability to meet user comfort	TBD

TEF Service: Mathematical verification		
	Name	***Description***
KPIs TEF Service implementation	Verification results delivered	The implementation delivers the verification results, in case no verification results can be obtained, it delivers and understandable error message and informs the operator.
	Verification setup received	The verification setup required for AI tool verification including all required data is received and checked. The verification setup contains a list of KPIs to be obtained during testing.
KPIs TEF Service & Business Use Case	Constraint verification for control decisions	The verification measures the violation of the AI tool against pre-defined bounds (e.g., optimization constraints).
	Distance to optimality	The verification measures how far the AI-tool control setpoints are from the optimal control setpoints.

TEF Service: Virtual facility access		
	Name	***Description***

KPIs TEF Servi ce	Accreditation	The service can provide access to the virtual environments needed for testing the AI tool
KPIs TEF Service &	Accreditation	AI vendors were able to access and use the virtualized environments, digital twins, or simulated infrastructures for AI testing.

TEF Service: Simulation-based testing		
	Name	***Description***
KPIs TEF Service implementation	Replicability and stress tests	The AI vendor is able to access a large number of scenarios, i.e., multiple LV grids to demonstrate the replicability of the tool.
	Testing	The AI vendor can successfully perform the simulation
	Performance assessment	Related to mathematical verification: how far the AI-tool control setpoints are from the optimal control setpoints
KPIs TEF Service & Business Use Case combination	Nr of violation over a period	Counts the number of violations over a pre-defined period.
	Violation severity	Absolute sum of voltages out of limits over a pre-defined period.
	Flexibility usage	Measures how much flexibility (i.e., power adjustments) was required to solve the technical violations, over a pre-defined period.

TEF Service: Experimental design for lab-based testing		
	Name	***Description***
KPIs TEF Service implem		
KPIs TEF Service & Business Use Case		

TEF Service: Facility access		
	Name	***Description***
KPIs TEF Servi ce	Accreditation	The service can provide access to the physical environment needed for testing the AI tool
KPIs TEF Servi ce &	Accreditation	AI vendors were able to access and use the physical infrastructure for AI testing in the laboratory

TEF Service: Lab-based testing		
	Name	***Description***
KPIs TEF Service implem	Replicability	Considering the same scenario representation, namely the grid and assets, the AI vendor is capable of conducting similar experiments as in the simulation-based testing.
KPIs TEF Service & Business Use Case combination	Nr of violation over a period	Counts the number of violations over a pre-defined period.
	Violation severity	Absolute sum of voltages out of limits over a pre-defined period.
	Flexibility usage	Measures how much flexibility (i.e., power adjustments) was required to solve the technical violations, over a pre-defined period.

TEF Service: Living lab testing

	Name	Description
KPIs TEF Service implem	Replicability	Considering the same scenario representation, namely the grid and assets, the AI vendor is capable of conducting similar experiments as in the simulation-based testing.
KPIs TEF Service & Business Use Case combination	Nr of violation over a period	Counts the number of violations over a pre-defined period.
	Violation severity	Absolute sum of voltages out of limits over a pre-defined period.
	Flexibility usage	Measures how much flexibility (i.e., power adjustments) was required to solve the technical violations, over a pre-defined period.

2.2 Hypotheses of TEF Test Use Case

Hypothesis	The performance of AI-based DER scheduling and control algorithms can be improved through data provision, domain-specific knowledge, performance assessment against benchmarks, and simulation and lab-based testing close to real-world conditions.
Success (scenario 1)	The performance of the algorithm improved when checked for typical key performance metrics (renewable energy self-consumption, voltage violations, among others) and when compared to existing baselines, given additional domain-specific data or knowledge in simulated and lab-based testing.
Unsuccessful (scenario 2)	It was not possible to improve the key performance metrics of the deployed algorithm through the provision or synthetization of data; It was not possible to compare the algorithm performance against any benchmark; It was not possible to provide domain-specific data; It was not possible to implement the simulation or lab-based testing for some case.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	Experiment workflow
Example: Data privacy/security (data synthetization) Text	Describe the workflow and the experiment conducted to evaluate that KPI. Example: Run the data synthetization on historical data and show that original data cannot be reverse-engineered. Text

2.4 Features of TEF Test Use Case

Method(s)	Optimization, Control
Platform	Developed by INESC TEC (based on Grid2Op but applied to the management of LV flexibility)

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
https://www.iso.org/committee/6794475/x/catalogue/
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. None

2.6 Challenges and issues

Challenges	Mitigation
Maintaining the quality of operational and external data for data provision.	Assure correct database management practices and correct handling of different operational data sources. Redundancy with additional and alternative data sources.
Ensuring quality data synthetization algorithms to assure real-world conditions.	Implement synthetic data quality metrics.

Selecting the right baselines for performance comparison	Start from common and simple models (e.g., linear regression). Literature review for state of the art models for comparison.
Development of the AI solution	-

2.7 Future Scope

Future Scope	
Enhancements	Additional TEF services can be considered. For instance, Conformity Assessment to ensure the AI solution meets industry standards and legal requirements.
Scalability	The execution of this test use case in other settings or LV grids require both voltage and power measurements from smart meters, which are not always available.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Integrating external data may increase the risk of exposing sensitive or proprietary information; Assure that data provision and synthetization do not lead to privacy breaches.
Sustainable Development Goals (SGD) to be achieved
7, 11

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	*Type*	*Reference*	*Status*	*Impact on use case*	*Originator / organization*	*Link*

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[7]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	*Tasks*	*Requirements for Data Exchange*
DSO (CEVE)	Data provider	Data cannot be exchanged directly because of security, must be anonymized.
AI Vendor (INESC TEC)	AI development	None
AI Vendor (Watt-IS)	AI development	Not willing to share AI tool source code.
Node host (INESC TEC)	TEF node host	

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic

[7] VILLASnode | VILLASframework

data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	***Description of the step***
Source	- DSO provides access to data that is stored regularly in a database (historical data). - AI vendor deploys trained AI tools in (or connects to) the node to use multiple services - -This data is made accessible to the AI vendor by the node host through APIs associated with the TEF node.
Time resolution	Less than one hour
Destination	- Node host receives data from DSO, and AI vendors receive the data to train the AI tool locally - The node host receives the trained AI tool or results from the AI vendor to perform benchmarking. - -Results of the services (e.g., benchmarking) are provided by the node host to the AI vendor

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	***Use***
HTTPS	Rest APIs will be able to interact with the services of this use case.

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

Tool 1: Edge load control for flexibility services

	Requirement description
Programming language	C/C++/Python.
Input (data format)	JSON | Includes participant-specific information, such as consumption and production forecasts, tariff details, and metadata about the equipment to be managed. It also considers community-related data, like energy sharing prices and forecasted shared energy, as well as flexibility requests, specifying the type, timing, required power, and compensation for the service.
Output (data format)	JSON | Specifies 15-minute intervals for turning equipment on or off, or, in the case of storage systems and EVs, charging or discharging at designated power levels. Additionally, it includes the response to any flexibility requests, indicating the power supplied if the request is accepted.
Interface	Protocol? API? Rest API
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Not intended
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Sensitivity analysis

Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Varying coefficient linear regression theory
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) N.A.
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Control, Linear optimization
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? Yes, verify bounds of control decisions and distance to optimal solution (ground-truth).

Tool 2: Voltage Control with distributed energy resources

Requirement description	
Programming language	C/C++/Python
Input (data format)	JSON | Voltage and active power injection measurements from smart meters and secondary substation; For reactive control → Real-time state of the system (state estimator); Available flexibility; For preventive control → forecasted state of the system, renewable energy forecasts, forecasted SoC of EVs connecting during the optimization horizon, forecasted arrival/departure of EVs.
Output (data format)	JSON | For both the control in real time (corrective), and a few hours ahead (preventive): Active power adjustments for flexible resources.
Interface	Protocol? API? Rest API
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Not intended
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning, Reinforcement learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Regression
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Linear regression, varying-coefficient (conditional parametric) models, privacy-preserving data exchange protocols, linear programming
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) N.A.
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Optimization, control
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? Yes, verify bounds of control decisions and distance to optimal solution (ground-truth).

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	*Requirement description*

3.6 Requirements for Data Synthetization

Usage of the TEF service "Data Synthetization" [8] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

	Requirement description
Need for synthetization	Data privacy or increased variety of data, but only as a service, not mandatory
Type of data	Load and RES generation data
Time details of data	Time series, 15-min to 1-hour resolution
Location of data synthetization	Synthetized within the Portuguese node
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	*Requirement description*
Missing values	Imputation of missing values.
Outlier detection	Outliers should be detected and handled.

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	*Requirement description*

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	*Requirement description*
Time constraint	For (near) real-time application, execution must be faster than data resolution

[8] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	***Task: Description***	***Objective***
	Data set: Get different data	Test AI algorithm with new data
	Real-time simulator: calculated setpoints by AI algorithm on a simulated grid	Verify that AI algorithm can cope with dynamic inputs/outputs

Validation of Load and RES forecasting algorithms

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.
Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	Application(s)	Name of Use Case
UC.1		Validation of Load and RES forecasting algorithms

1.2 Version management

Version Management			
Version No.	Date	Name of Author(s)	Changes
0.1	20.01.2025	Carlos Silva	First draft
0.2	19.02.2025	Carlos Silva	Add Sections 2 and 3
0.3	15.05.2025	Carlos Silva	Adapt to new template

1.3 Business Use Case(s)

Priority	Business Use Case	Scope Description of the Business Use Case
1	Load and RES generation forecasting	Load and RES generation AI-based forecasting algorithms to support energy management and decision-making in households, energy communities, and electric vehicle charging stations.
2	Forecasting electric vehicle charging patterns	Load AI-based forecasting algorithms to support energy management and decision-making in electric vehicle charging stations.

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	Validating and improving the performance of AI-based Load and RES forecasting algorithms for energy management
Objective(s)	(6) Development of an environment where TEF users and AI developers can enhance and rigorously test their forecasting models, both for Load and RES generation, based on both general and domain-specific data and knowledge. (7) Ensure that users can access operational and external data, and relevant features to improve their solutions, by providing access to relevant data sources, tools, and libraries. (8) Ensure that users can thoroughly test their solutions, by providing access to a pipeline consisting of performance assessment or evaluation tools and baseline models for benchmarking. (9) Provide details on the training process of the AI solution, such as explainability techniques for analysis of feature importance, and computing the energy consumption related to the model training. (10) This use case is expected to benefit providers of AI solutions directly, by improving the development and testing phase of their models. As these forecasting algorithms are mainly targeting predictive energy management, it is expected that the use case indirectly improves other stakeholders that use the forecasting models for decision-making (e.g., electricity providers' market participation, network operators' congestion management, or end-users' demand-side management).

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description
The use of the TEF relates mainly to its data provision and performance evaluation capabilities and services. This use case not only improves, tests, and validates AI models that relate to the forecasting of load and RES generation, but also ensures they are evaluated under realistic and diverse settings, even under conditions of low data availability through data synthetization. This supports the readiness of AI solutions for deployment, specifically for real-world implementations of forecasting models, and for solutions within energy systems that require predictive information. More specifically, using the TEF allows this use case to indirectly enhance the

energy management and resource optimization capabilities at multiple levels (households, energy communities, electric vehicle charging stations), since they depend greatly on the performance of forecasting applications.
Complete description
Load and RES forecasting algorithms are critical for applications in energy systems, since many entities require these inputs for decision-making, namely, to compute control actions and predictive management strategies. Using a pipeline of TEF services can enhance the performance of the algorithms and make them more robust in real-world deployment. The expected use of the TEF is the following: (1) The user deploys the load or RES forecasting algorithm in the TEF. (2) The user can request domain-specific data (operational and external) from the TEF, to support model training and complement the available proprietary dataset. Alternatively, the user can request synthetized data, ensuring data privacy concerns are met. (3) The user can request relevant additional features from a Knowledge Store, to improve model training with domain-specific knowledge, features, and feature transformations. (4) The user can request a performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks. (5) The user can check the importance of features used for training with explainability techniques and get insight of energy consumption related to the model's training process. (6) The user can request a mathematical verification, which can detect abnormal outputs and the occurrence of unfeasible or unrealistic outcomes. (7) The user refines the model iteratively with the additional data, features, and performance evaluation information, and applies it to the business use case.
Stakeholders
Providers of energy analytics and management tools; network operators; electricity suppliers; customers/end-users (e.g., residential electricity consumers, energy community participants).
Stakeholders' assets, values
Competitiveness of energy management solutions, data privacy.
System's threats and vulnerabilities
Forecasting models that are not robustly evaluated and tested, or that have a low performance, can lead to sub-optimal or incorrect decision-making due to poor predictive information provided to energy management solutions. This directly affects, for instance, control actions in consumer households and energy communities, which can affect energy costs and other key performance indicators. Data privacy of consumers is also a threat that must be accounted for.

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data provision	Domain-specific data (operational and external) to support model training and complement the available dataset.
2	Data Synthetization	Synthetic data of load and renewable energy generation to ensure data privacy concerns are met.
2	Knowledge store	Relevant additional features from a Knowledge Store, to improve model training with domain-specific knowledge, features, and feature transformations.
2	AI Tool Benchmarking	Performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.
5	Feature importance analysis	Explainability techniques to check the importance of features for model training.
6	Mathematical verification	Detect abnormal, unfeasible or unrealistic outputs from the AI model.
7	Energy consumption of AI algorithm	Monitoring and evaluating the energy usage of AI algorithms during training to support the development of sustainable AI models.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***

Performance evaluation of recommendation algorithm for energy efficiency	
Performance evaluation of energy-sharing mechanism	
Testing of DER scheduling/control algorithms	
Performance evaluation of network monitoring algorithm	

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	*Use case description (short)*
AI-powered virtual energy management	Improve energy efficiency in households based on smart metering data with load disaggregation and tailored recommendations.
Optimization of energy community operation	Management of energy communities including AI-powered features such as optimal scheduling of community storage and optimal sharing coefficients for participants.
Voltage Control with distributed energy resources	Algorithms for voltage control of flexible distributed energy resources based on AI models for state estimation.
Edge load control for flexibility services	Provide flexibility services with control of flexible loads (Storage, EWH, Heat pumps) based on edge AI and cloud solutions.
Network topology estimation	AI-based models to infer the topology of the electrical grid and fix misspecified grid configurations (e.g., consumer phases).
State estimation and network quality	AI-based state estimation algorithms to improve observability and control of the electrical grid, including network quality metrics.

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	*TEF service description (short)*
Data tracing	Trace the flow and transformation of data to ensure transparency and security.
Conformity assessment	Evaluation of AI solutions to ensure they meet industry standards, legal requirements, or safety regulations.

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Data Provision		
	Name	***Description***

KPIs TEF Service implementation	Data access	The AI vendor can receive data from the TEF to train the AI model
KPIs TEF Service & Business Use Case	Load and RES generation data access	Load and RES generation data from the DSO are provided to the AI vendor via TEF on demand

TEF Service: Data Synthetization		
	Name	***Description***
KPIs TEF Service implementation	Data privacy	Original data cannot be reverse-engineered from synthetic data.
	Data access	The AI vendor can receive synthetized data from the TEF to train the AI model
	Operation modes	Synthesizer module can provide different types of data
KPIs TEF Service & Business Use Case	Synthetized Load and RES generation data access	The AI vendor can receive synthetized load and RES generation data from the DSO via TEF on demand
	Synthetic data quality metrics	Similarity between synthetized load and RES generation data and real data (e.g., time-series based properties).

TEF Service: Knowledge Store		
	Name	***Description***
KPIs TEF Service implementation	Access to domain-specific features	The service can provide a set of features that are domain-specific based on the input data
	Multiple domains	The service can handle more than 1 domain (e.g., wind, solar PV generation)
KPIs TEF Service & Business Use Case	Number of domain-specific features	Measure the number of features provided by the service to the load and RES generation AI forecasting tool
	Forecasting metric improvement	Load and RES generation AI forecasting tool metrics (e.g., MSE, MAE) should improve when domain-specific features are provided

TEF Service: AI tool benchmarking		
	Name	***Description***
KPIs TEF Service implementation	Benchmark comparison	A baseline/state-of-the-art AI tool or ranking list is given to compare the results of the benchmarked AI tool and rank its performance.
	Data flow	The service is accessible by the AI vendor and necessary data can be shared. This includes the trained AI tool and benchmarking results.
KPIs TEF Service & Business Use Case	Load forecasting metrics	Load AI forecasting tool metrics (e.g., MSE, MAE) compared to other models
	RES generation forecasting metrics	RES generation AI forecasting tool metrics (e.g., MSE, MAE) compared to other models

TEF Service: Feature Importance Analysis		
	Name	***Description***
KPIs TEF Service	Shapley value data table delivered	Data table with Shapley values is available and successfully delivered to the user

KPIs TEF Service & Business Use Case	Shapley values	Shapley values for each input feature, and input feature combination, is obtained for the Load and RES generation forecasting tool, explaining how much each input feature is responsible to arrive at the final prediction.

TEF Service: Energy consumption of AI algorithm		
	Name	***Description***
KPIs TEF Service implementation	Environment for AI model training	The service can provide a controlled environment where the AI tools are deployed and monitored
	Energy consumption measurements	The service can measure precisely the energy consumption during the training process
KPIs TEF Service &	Energy consumption	The AI vendor receives the results of energy consumption due to training the load and RES generation forecasting models

TEF Service: Mathematical Verification		
	Name	***Description***
KPIs TEF Service implementation	Verification results delivered	The implementation delivers the verification results, in case no verification results can be obtained, it delivers and understandable error message and informs the operator.
	Verification setup received	The verification setup required for AI tool verification including all required data is received and checked. The verification setup contains a list of KPIs to be obtained during testing.
KPIs TEF Service &	Bound verification for load and RES generation)	The verification measures the violation of the AI tool against pre-defined bounds (e.g., physical constraints or bounds, maximum capacities).

2.2 Hypotheses of TEF Test Use Case

Hypothesis	The performance of AI-based forecasting algorithms can be improved through data provision, domain-specific knowledge, and robust performance assessment against benchmarks.
Success (scenario 1)	The performance of the forecasting algorithm improved when checked for typical error metrics and when compared to existing baselines, given additional domain-specific data or knowledge. Additionally, improved forecasting performance led to enhanced decision-making when integrated with an energy management solution that has a set of pre-defined key performance indicators.
Unsuccessful (scenario 2)	It was not possible to improve the forecasting error metrics of the deployed algorithm through the provision or synthetization of data, or through the provision of additional domain-specific features; It was not possible to compare the algorithm performance against any benchmark; It was not possible to provide domain-specific data.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
Forecasting metric improvement	Calculate common forecasting performance metrics for the AI tool, after providing different services (e.g., data provision and features from the Knowledge Store), and compare it with previous metrics and other benchmark models.

2.4 Features of TEF Test Use Case

Method(s)	Prediction
Platform	None

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
Identify here relevant standards for the use case. A good source of information: https://www.iso.org/committee/6794475/x/catalogue/
Standardization requirements
Descriptions of standardization opportunities/requirements that are derived from the use case. None

2.6 Challenges and issues

Challenges	***Mitigation***
Maintaining the quality of operational and external data for data provision	Assure correct database management practices and correct handling of different operational data sources. Redundancy with additional and alternative data sources.
Ensuring quality data synthetization algorithms to assure real-world conditions	Implement synthetic data quality metrics.
Selecting the right baselines for performance comparison	Start from common and simple models (e.g., linear regression). Literature review for state of the art models for comparison.

2.7 Future Scope

Future Scope	
Enhancements	Operationalize the forecasting tools with simulation-based tools to compute model performance on real conditions. Additional TEF services can be considered. For instance, Conformity Assessment to ensure the AI solution meets industry standards and legal requirements. Ensure that all kinds of forecasting tools are possible to use within the TEF.
Scalability	Stress-testing environment for forecasting applications, simulating extreme weather conditions and assessing forecasting model performance due to climate change impacts.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Integrating external data may increase the risk of exposing sensitive or proprietary information; Assure that data provision and synthetization do not lead to privacy breaches.
Sustainable Development Goals (SGD) to be achieved
7, 11

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[9]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	*Tasks*	*Requirements for Data Exchange*
DSO (CEVE)	Data provider	Data cannot be exchanged directly because of security, must be anonymized.
AI Vendor (INESC TEC)	AI development	None
AI Vendor (Watt-IS)	AI development	Not willing to share AI tool source code.
Node host (INESC TEC)	TEF node host	

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	*Description of the step*
Source	- DSO provides access to data that is stored regularly in a database (historical data). - AI vendor deploys AI tools in the node to use multiple services - This data is made accessible to the AI vendor by the node host through APIs associated with the TEF node.
Time resolution	Historical data, no time constraints
Destination	- Node host receives data from DSO, and AI vendors receive the data to train the AI tool - Results of the services are provided by the node host to the AI vendor

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	*Use*
HTTP	Communication of DSO and AI Vendor with Node host

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

	Requirement description
Programming language	Python
Input (data format)	CSV, JSON
Output (data format)	CSV, JSON

[9] VILLASnode | VILLASframework

Interface	
Sharing the tool	Some tools available via Git, others sharing not intended
AI Category	Supervised learning
Goal	Regression
Model type	Tree-based, Neural networks, Linear models, Ensemble methods
Domain	Time-series and tabular
Package	Scikit-learn
Problem nature	Regression problem
Verification	Yes, verify bounds for load and generation

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	***Requirement description***
Service compatibility	AI model tested should be compatible with the requirements imposed by services such as Feature Importance Analysis and Mathematical Verification

3.6 Requirements for Data Synthetization

Usage of the TEF service “Data Synthetization” [10] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

	Requirement description
Need for synthetization	Data privacy or increased variety of data, but only as a service, not mandatory
Type of data	Load and RES generation data
Time details of data	Time series, 15-min to 1-hour resolution
Location of data synthetization	Synthetized within the Portuguese node
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	***Requirement description***
None	

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	***Requirement description***
Computational infrastructure	For testing AI algorithms and related benchmark models

3.9 Time Constraints and Time Frame

[10] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	***Requirement description***
None	

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***
Anonymization	Assured by the data provider

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	***Task: Description***	***Objective***

Performance evaluation of network monitoring algorithms

0 Common Terms and Definitions

Follow the AI terminology and taxonomy that is currently being harmonized between EU and U.S. https://digital-strategy.ec.europa.eu/en/library/eu-us-terminology-and-taxonomy-artificial-intelligence

AI-EFFECT Terms and Definitions	
Term	**Definition**
TEF(Testing and Experimentation Facility)	A facility designed to support AI development and validation by providing a structured environment for testing and experimentation. It consists of multiple **Nodes** and offers **TEF Services**. Its primary function is to facilitate testing processes that improve the technical readiness of AI solutions.
TEF Services	Specialized offerings from the TEF to support AI development. These services are broader than Test Use Cases and include offerings like data simulation or data provision (access to quality, domain-specific data) which alone do not constitute a Test Use Case. These Services may also include conformity assessments (for regulatory and technical compliance), AI model evaluations (to test model performance), designing controlled testing environments (for structured end-to-end AI tests), and defining necessary infrastructure (both physical and digital assets).
TEF Test Use Case	A structured test process within the Testing and Experimentation Facility (TEF) specifically aimed at validating and experimenting with a Business Use Case. Unlike general Business Use Cases, TEF Test Use Cases focus on testing procedures, involving end-to-end steps for verifying AI applications in a controlled environment. The Test Use Cases can include TEF Services.
Business Use Case	A specific process within an organization, characterized by data-driven inputs and outputs and which have potential to be improved or automated by the application of AI. An example in the energy sector business might be "Electricity power flow management," where data science techniques such as machine learning or could optimize or automate the process of powerflow.

Node	A specific part of the TEF that provides the necessary technology (both physical and digital components) to support a Test Use Case. Nodes may include physical components, such as batteries or transformers, and digital components, like specialized simulators or software tools, or datasets. Each node serves as a centre for innovation, focusing on testing, experimentation, and the co-creation of AI solutions related to its supported use cases.

1 Description of the TEF Test Use Case

1.1 Name of the TEF Test Use Case

ID	Application(s)	Name of Use Case
UC.3		Performance evaluation of network monitoring algorithms

1.2 Version management

Version Management			
Version No.	Date	Name of Author(s)	Changes
0.1	26.02.2025	Carlos Silva	First draft
0.2	05.03.2025	Gil Sampaio	Contributions to the initial description
0.3	16.05.2025	Carlos Silva	Adapt to new template

1.3 Business Use Case(s)

Priority	Business Use Case	Scope Description of the Business Use Case
1	State estimation and network quality	AI-based state estimation algorithms to improve observability and control of the electrical grid, including network quality metrics
2	Network topology estimation	AI-based models to infer the topology of the grid, including electrical characteristics if cables and lines, and fix erroneous grid configurations

1.4 Scope and objectives of the TEF Test Use Case

Scope and Objectives of TEF Test Use Case	
Scope	Validate and evaluate the performance of AI algorithms for state estimation, network quality improvement, and network topology estimation. The focus is on testing these algorithms in a controlled environment to assess their accuracy.
Objective(s)	(11) Development of an environment where TEF users and AI developers can enhance and rigorously test their monitoring algorithms, namely: a. State estimation (voltages and/or powers) under limited real-time observability. b. Grid topology discovery or correction. c. Connection phase identification/assignment of single-phase installations. d. Electrical characteristics correction of cables and lines. (12) Ensure that users can access operational and external data by providing access to relevant data sources and tools. (13) Ensure that users can enhance their solutions, by providing access to a pipeline consisting of performance assessment or evaluation tools, and baseline models for benchmarking. (14) Provide testing environments for the AI-based network monitoring algorithms, namely simulated digital environments. (15) This use case is expected to benefit providers of AI solutions directly, by improving the development phase of their model, namely exposing them to simulated testing environments that are close to real-world conditions. It is also expected that the use case indirectly improves other stakeholders that use these solutions for decision-making (e.g., network operators).

1.5 Narrative of TEF Test Use Case

Narrative of TEF Use Case
Short description

The use of the TEF relates mainly to its data provision, benchmarking, and testing capabilities in digital environments built as replicas of real systems. This use case not only improves and validates AI models that relate to system state estimation and topological and electrical characterization in LV grids, but also creates value for AI solution developers by ensuring that their models are tested in different kinds of environments (simulation-based experiments) to support and validate their deployment in real-world applications. Consequently, using the TEF will allow us to improve observability and control of the electrical grid, including network quality metrics.

Complete description

Different challenges are now found and envisioned at the lower levels of the distribution system: the growing penetration of renewable resources and storage devices, including electric vehicles, demand-side-management strategies, the establishment of microgrids with independent controllability and islanding capabilities, p2p markets, and so on.

The development of monitoring tools is of paramount importance to ensure that the operator's situational awareness is adequate. A number of challenges can easily be identified:

- Despite the widespread deployment of smart meters, technical limitations in the communication infrastructure often prevent real-time monitoring of the grid.
- Grid topology and/or the electrical characteristics of lines and cables are frequently missing or inaccurate, hindering the use of conventional power flow-based tools or any optimisation algorithms that rely on a physical model of the grid.
- Records of phase assignment for single-phase customers are often incorrect and may change over time due to grid expansions or interventions. This severely compromises efforts to balance grid loads and may reduce the grid's hosting capacity for both new loads and distributed generation.

In the case of state estimation algorithms, the objective is to provide the most likely state of the network, for the real time or for the next few hours (i.e., predictive), relying exclusively on data (e.g., measurements from smart meters) and without resorting to grid topology or electrical characteristics. Despite the widespread of smart meters, the reliance on Power Line Communication constraints the real time monitoring, with many configurations where measurements reach centralized systems only once per day. The application of state estimation in LV grids is of particular interest in this scenario where techniques rely on a small number of smart meters that communicate in real-time, or other relevant data known in real time, to exploit patterns in historical data and reconstruct the system state. Figure 1 illustrates the rationale behind a state estimation methodology that searches for analogous past events in order to reconstruct the system state.

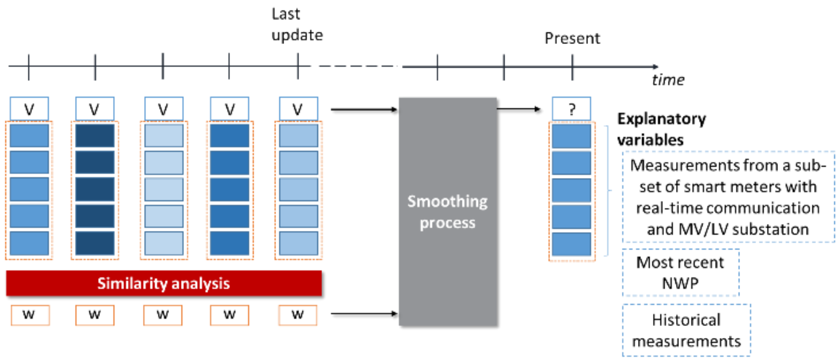


Figure 9: Illustration of a method that searches for similar past events to estimate the most likely value of a measurement.

To test a state estimation algorithm, the following conditions will be considered:

- Historical dataset of voltages and active power of each consumer.
- Real-time measurements (voltage and/or active power) from a subset of meters.
- Real-time measurements (voltage and/or active power) at the secondary substation.
- Historical dataset of irradiance.
- Real-time irradiance.

For different purposes, such as grid expansion planning or to enable power-flow based algorithms, it is necessary to have a rigorous model of the grid (topology and electric characteristics of cables). This information is many times inexistent or incorrect for LV grids, and the huge volumes of data gathered by the smart meters can be used to develop discovery or corrective approaches. Figure 2 shows the results obtained using a methodology that leverages smart meter data within an evolutionary algorithm to identify the correct grid topology and cable impedances.

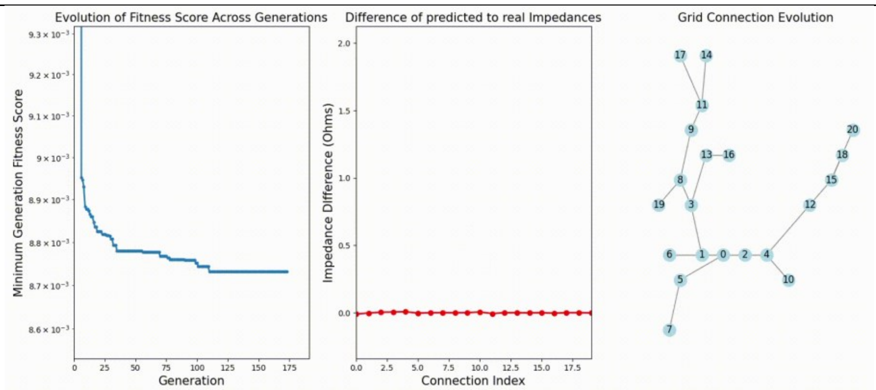


Figure 10: Illustration of the evolution of a Genetic Algorithm to find the correct topology and correct cable impedance.

Finally, accurate phase identification is vital for utility companies as it enhances grid management by enabling better load balancing across phases, which can reduce energy losses and prevent overloads. Additionally, precise phase data supports more accurate assessments of how much more DER the system can handle.

Therefore, precise models and network monitoring algorithms are critical for applications in energy systems. For instance, the safe operation of the electrical grid depends on the ability to observe and monitor its real-time state and compute the appropriate control actions. Using a pipeline of TEF services can enhance the performance of the algorithms and make them more robust in real-world deployment through access to complex testing environments. The expected use of the TEF is the following:

(7) The user deploys the network monitoring algorithm in the TEF.
(8) The user can request domain-specific data (operational and external) from the TEF, to support model training and complement the available proprietary dataset. Alternatively, the user can request synthetized data, ensuring data privacy concerns are met.
(9) The user can request a performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.
(10) The user can request a mathematical verification, to identify critical operating regimes for the AI tool.
(11) The user can request a simulation-based test, which will result in access to a simulated digital testing environment with the necessary tools to test the AI tool under diverse conditions close to real-world applications.

Stakeholders
Grid operators, consumers, producers.
Stakeholders' assets, values
Privacy
System's threats and vulnerabilities

1.6 TEF Services

List the TEF services that will be demonstrated with the node. Stick to the defined TEF services and provide information about the TEF Services relate to the TEF Test Use Case. Rank them according to their priority.

Priority	***TEF Service***	***Objective of the service applied to the TEF Test Use Case***
1	Data provision	Domain-specific data (operational) to support model training and complement the available dataset
3	Data Synthetization	Synthetic data to ensure data privacy concerns are met
2	AI Tool Benchmarking	Performance evaluation of the model from the TEF, which can include a comparison with baseline models and available benchmarks.
1	Virtual Facility Access	Provide access to virtual assets for simulation-based testing of network monitoring algorithm
1	Simulation-based testing	Network monitoring algorithms are evaluated under a set of diverse operating conditions in a simulated digital environment.

1.7 Other possible TEF Test Use Cases

List other Test Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***

Performance evaluation of recommendation algorithm for energy efficiency	
Performance evaluation of energy-sharing mechanism	
Validation of forecasting algorithms	
Testing of DER scheduling/control algorithms	

1.8 Other possible Business Use Cases

List other Business Use Cases that would be possible with the node, and interesting to the stakeholders involved in the node.

Use case name	***Use case description (short)***
AI-powered virtual energy management	Improve energy efficiency in households based on smart metering data with load disaggregation and tailored recommendations.
Optimization of energy community operation	Management of energy communities including AI-powered features such as optimal scheduling of community storage and optimal sharing coefficients for participants.
Edge load control for flexibility services	Provide flexibility services with control of flexible loads (Storage, EWH, Heat pumps) based on edge AI and cloud solutions.
Voltage Control with distributed energy resources	Algorithms for voltage control of flexible distributed energy resources based on AI models for state estimation.
Forecasting electric vehicle charging patterns	AI-based models for prediction of EV charging sessions to support control of flexible resources.
Load and RES generation forecasting	AI-based models for load and RES generation forecasting.

1.9 Other possible TEF Services

List other TEF services that would be possible with the node, and interesting to the stakeholders involved in the node.

TEF service name	***TEF service description (short)***
Conformity Assessment	Evaluation of AI solutions to ensure they meet legal requirements (e.g., AI Act) and safety regulations, potentially framed in a checklist

1.10 Other remarks

Describe other remarks important to communicate to the TEF developers about this use case

2 Detailed Definition of TEF Test Use Case

2.1 Key Performance Indicators (KPI)

KPIs need to be defined on two levels: First, KPIs evaluating the performance of the TEF Service implementation are defined. Second, KPIs evaluating the performance of the TEF Service on a specific Business Use Case are defined. These specific KPIs should make the TEF Test Use Case more concrete and easier to understand for the development of the TEF. The description specifies the KPI and may include specific and the calculation of these targets.
Please fill in a copy of the table for each TEF Service.

TEF Service: Data Provision		
	Name	***Description***
KPIs TEF Service	Data access	The AI vendor can receive data from the TEF to train the AI model

KPIs TEF Service & Business Use Case	Data access	Metering data from the DSO are provided to the AI vendor via TEF on demand to support training of network monitoring algorithm
	Data availability	Historical metering data from the DSO is enough to train the AI model

TEF Service: Data Synthetization		
	Name	***Description***
KPIs TEF Service implementation	Data privacy	Original data cannot be reverse-engineered from synthetic data.
	Data access	The AI vendor can receive synthetized data from the TEF to train the AI model
	Operation modes	Synthesizer module can provide different types of data
KPIs TEF Service & Business Use Case	Data access	Synthetized metering data from the DSO are provided to the AI vendor via TEF on demand to support training of network monitoring algorithm
	Synthetic data quality metrics	Similarity between synthetized metering data and real data (e.g., time-series based properties).

TEF Service: AI Tool benchmarking		
	Name	***Description***
KPIs TEF Service implementation	Benchmark availability	A set of baseline/state-of-the-art AI tool are available to compare the results of the benchmarked AI tool and rank its performance.
	Data flow	The service is accessible by the AI vendor and the necessary data can be shared. This includes the trained AI tool and benchmarking results.
KPIs TEF Service & Business Use Case	State estimation metrics	Measures how good the estimates are with respect to the real values (e.g., power, voltage), and compared against baselines
	Topology estimation metrics	Measures how good the estimates are with respect to the real values (e.g., impedance matrix), and compared against baselines

TEF Service: Virtual facility access		
	Name	***Description***
KPIs TEF Service	Accreditation	The service can provide access to the virtual environments needed for testing the AI tool
KPIs TEF Service & Business Use Case	Accreditation	AI vendors were able to access and use the virtualized environment for testing of state estimation tool.
	Accreditation	AI vendors were able to access and use the virtualized environment for testing of state estimation tool.

TEF Service: Simulation-based testing		
	Name	***Description***
KPIs TEF Service implementation	Access to environment	The AI vendor can interact with the digital environment to test the AI model
	Testing	The AI vendor can successfully perform the simulation
	Performance assessment	Reports the simulations results, i.e., measures how good the estimates are with respect to the real values
KPIs TEF Service & Business Use Case	State estimation metrics	Reports the simulations results, i.e., measures how good the estimates are with respect to the real values (e.g., power, voltage)
	Topology estimation metrics	Reports the simulations results, i.e., measures how good the estimates are with respect to the real values (e.g., impedance matrix), and compared against baselines

2.2 Hypotheses of TEF Test Use Case

Hypothesis	The performance of AI-based network monitoring algorithms can be improved through data provision, robust performance assessment against benchmarks, and simulation-based testing.
Success (scenario 1)	The performance of the network monitoring algorithm improved when checked for typical error metrics and when compared to existing baselines, given additional domain-specific data. Additionally, the improved performance led to enhanced decision-making when integrated with subsequent management or control solution that has a set of pre-defined key performance indicators.
Unsuccessful (scenario 2)	It was not possible to improve the error metrics of the deployed algorithm through the provision of data; It was not possible to compare the algorithm performance against any benchmark; It was not possible to provide domain-specific data.

2.3 Experiment demonstrating the functionality of the TEF Test Use Case

KPI Name (TEF Service)	***Experiment workflow***
State estimation metrics improvement	Calculate common network monitoring performance metrics for the AI tool, after providing different services (e.g., data provision), and compare it with previous metrics and other benchmark models.

2.4 Features of TEF Test Use Case

Method(s)	Prediction
Platform	Custom environment developed by INESC TEC (based on Grid2Op but applied to the management of LV grids).

2.5 Standardization opportunities and requirements

Classification Information
Relation to existing standards
https://www.iso.org/committee/6794475/x/catalogue/
Standardization requirements
None

2.6 Challenges and issues

Challenges	***Mitigation***
Maintaining the quality of operational and external data for data provision	Assure correct database management practices and correct handling of different operational data sources. Redundancy with additional and alternative data sources.
Selecting the right baselines for performance comparison	Start from common and simple models (e.g., linear regression). Literature review for state of the art models for comparison.

2.7 Future Scope

Future Scope	
Enhancements	Include Knowledge store service for feature provision, for instance related to numerical weather predictions.
Scalability	The execution of this test use case in other settings or LV grids require voltage measurements, which are not always available.

2.8 Societal concerns (ethical concerns)

Societal concerns
Description
Integrating external data may increase the risk of exposing sensitive or proprietary information; Assure that data provision and synthetization do not lead to privacy breaches.
Sustainable Development Goals (SGD) to be achieved

7, 11

2.9 References of TEF Test Use Case

References (**reports, mandates and regulatory constraints, papers, patents, press releases**) associated with the TEF Test Use Case and that support interest from industry and/or regulatory bodies or provide additional information from past trials/ideas. Furthermore, identify any European legal issues that might affect the design and requirements of the function, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

References						
No.	***Type***	***Reference***	***Status***	***Impact on use case***	***Originator / organization***	***Link***

3 TEF Platform requirements for algorithms, data, infrastructure, protocols, VILLASnode

The following points concentrate on the technical requirements of the TEF nodes especially regarding VILLASnode[11]. The aim is to provide some guidelines for the possible integration of VILLASnode.

3.1 Partners involved within a node

Within a TEF node, different partners can be involved. Who provides the data, who develops the AI tool, who hosts the node? What security is required to exchange data between them?

Function of Involved Partners	***Tasks***	***Requirements for Data Exchange***
DSO (CEVE)	Data provider	Data cannot be exchanged directly because of security, must be anonymized.
AI Vendor (INESC TEC)	AI development	None
Node host (INESC TEC)	TEF node host	

3.2 Narrative of data flow

An idea of the data flow helps to be aware of the needed requirements and to get yourself a high-level understanding and to help others to understand it. Although some things might be doubled with the next points, this should be seen as an overview. What is the source and destination? Is it a real-time data stream or historic data? Are there time constraints? Which machine can you use (Linux, VM) or do you prefer to get a pre-configured Raspberry Pi? Include graphics if needed.

Step of data flow	***Description of the step***
Source	- DSO provides access to data that is stored regularly in a database (historical data). - AI vendor deploys trained AI tools in (or connects to) the node to use multiple services -This data is made accessible to the AI vendor by the node host through APIs associated with the TEF node.
Time resolution	Less than one hour
Destination	-Node host receives data from DSO, and AI vendors receive the data to train the AI tool locally -The node host receives the trained AI tool or results from the AI vendor to perform benchmarking. -Results of the services (e.g., benchmarking) are provided by the node host to the AI vendor

3.3 Requirements for communication protocols in node

Between the involved partners of single node, data exchange is realized by communication protocols, e.g., WebRTC or UDP. But are you planning to use specific ones? For what do you use the protocol? If you use

[11] VILLASnode | VILLASframework

hardware, by which protocol is the hardware interfaced? Do you expect any interoperability issues? Please indicate what you already know or expect.

Protocol	Use
HTTPS	Rest APIs will be able to interact with the services of this use case.

3.4 AI tool

The AI tool subject to the use case has specifications that are necessary to know for the project. This part should specify more detailed technical information about the AI tool/algorithm. Which programming language? Needs the algorithm to be started from command line? Can the algorithm be containerized? Should the algorithm run locally or in a cloud? How can the algorithm be interfaced (protocol, API)? What are the inputs and outputs?

Tool 1: State Estimation and network quality

Requirement description	
Programming language	C/C++/Python
Input (data format)	JSON | Voltage and active power injections measurement from smart meters including secondary substation (historical data); Real-time data includes active power flow and voltage measurements at the secondary substation and a subset of the smart meters.
Output (data format)	JSON | Real-time state of the system including estimated voltage magnitudes and active power injections for all smart meters that do not communicate in real time
Interface	Protocol? API? Rest API
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Not intended
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning; supervised learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Estimation
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Kernel smoothing, kernel density estimation, metaheuristics
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) N.A.
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Deterministic and probabilistic estimation with optimization of hyperparameters
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? No

Tool 2: Network topology estimation

Requirement description	
Programming language	Python
Input (data format)	JSON | Voltage and active power injections measurement from smart meters including secondary substation (historical data); Real-time data includes active power flow and voltage measurements at the secondary substation and a subset of the smart meters.
Output (data format)	JSON | Depends on the objective: phase where most likely a single-phase customer is connected to; estimated topology of the grid; impedances of lines
Interface	Protocol? API? Rest API
Sharing the tool	Provide via Git or sharing not intended, should only run locally in node Not intended
AI Category	Choose the most appropriate learning method for your model (Supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, statistical learning) Statistical learning
Goal	Specify the task your AI is solving (for example; classification, regression, clustering, anomaly detection, dimensionality reduction) Estimation
Model type	What type of model does your AI use (neural network, tree-based, probabilistic model, distance based, linear model, ARMA model, ensemble methods Genetic algorithm, linear regression, hierarchical clustering algorithms, probability metric measurements.
Domain	Specify the type of data your model will be using (Time-series models, graph-based models, tabular models, image models, text models) Time-series, graph-based models
Package	Which package is the AI tool made in? (Scikit-learn, TensorFlow/Keras, Pytorch) TBD
Problem nature	Describe the underlying problem the AI tool addresses? Is it a classification problem? Is it a control problem? Is it an optimization? And what kind of objective and constraints is the problem made of? Does it have integers, or only continuous variables? Does it have linear, quadratic, or non-linear constraints? Topology and parameters estimations
Verification	Do you want to verify whether a constraint is violated over the entire input region, and what kind of constraint would that be? A bound, a physical equation? Do you want to verify the distance to the optimal solution i.e. the ground truth? Do you want to identify adversarial examples? Do you want to identify a robust input region? No

3.5 Requirements for testing/validating AI algorithms and models for the use case

This part should provide information if requirements are needed to test and validate the AI tool.

Requirement name	Requirement description
Computational requirements	To be determined

3.6 Requirements for Data Synthetization

Usage of the TEF service "Data Synthetization" [12] for power grid data may be needed to support the use case. Data synthetization is the generation of synthetic data based on historical data or metadata. What requirements are there? If you are not planning to use the data synthesizer, you do not need to fill anything.

[12] In the context of the AI EFFECT project, a tailored synthesizer will be developed for TSO grid data, where the aim is to make the synthesizer adaptable to other use cases that involve similar type data and requirements, however, post-processing will be needed.

	Requirement description
Need for synthetization	Data privacy or increased variety of data, but only as a service, not mandatory
Type of data	Time series
Time details of data	< 1h hour resolution, minimum 2 months historical data
Location of data synthetization	Synthetized within the Portuguese node
Other details	

3.7 Requirements for Data Processing

Data exchange within a node depends on the original format of the data, where and how it should go, and which manipulation of data is required. Which data format is required for input and output of the AI tool? Needs the data manipulation, e.g., timestamps, calculation of offset, …

Requirement name	***Requirement description***
Missing values	Imputation of missing values
Outlier detection	Outliers should be detected and handled.

3.8 Requirements for Infrastructure

This part provides more general requirements for the hardware/software infrastructure which could not be provided in the previous parts. It also includes ideas for the future, e.g., integration of a data space.

Requirement name	***Requirement description***
TBD	

3.9 Time Constraints and Time Frame

Use cases might have time constraints. How fast data should be exchanged? Also, the time frame of the use case is interesting. How long will the use case run (during demonstration)?

Requirement name	***Requirement description***
Time constraint	Execution must be faster than data resolution

3.10 Security constraints

Do you have security concerns when sending the data? Do you have requirements for encrypting the data additionally? Or is the data privacy solved by synthesizing the data?

Requirement name	***Requirement description***
None	

3.11 Partners involved between nodes

Between TEF nodes, different partners can be involved. For example, the advantage would be to train with different data, to test own data with different algorithms, to test algorithms on locally unavailable physical devices or simulators. The column involved partners do not need to be filled out if it is unclear yet who can provide it.

Involved Partners	***Task: Description***	***Objective***

References

[1] R. Bessa, M. Yagoubi and M. Leyli-abadi, "D1.1 AI4REALNET framework and use cases," 2024.

[2] S. Vogel, M. Mirz, L. Razik and A. Monti, "An open solution for next-generation real-time power," in *2017 IEEE Conference on Energy Internet and Energy System*, Beijing, 2017.

[3] FEIN Aachen e.V., "VILLASframework," Aachen.

Acknowledgements

AI-EFFECT is supported by the European Union's Horizon Europe programme under agreement 101172952. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.