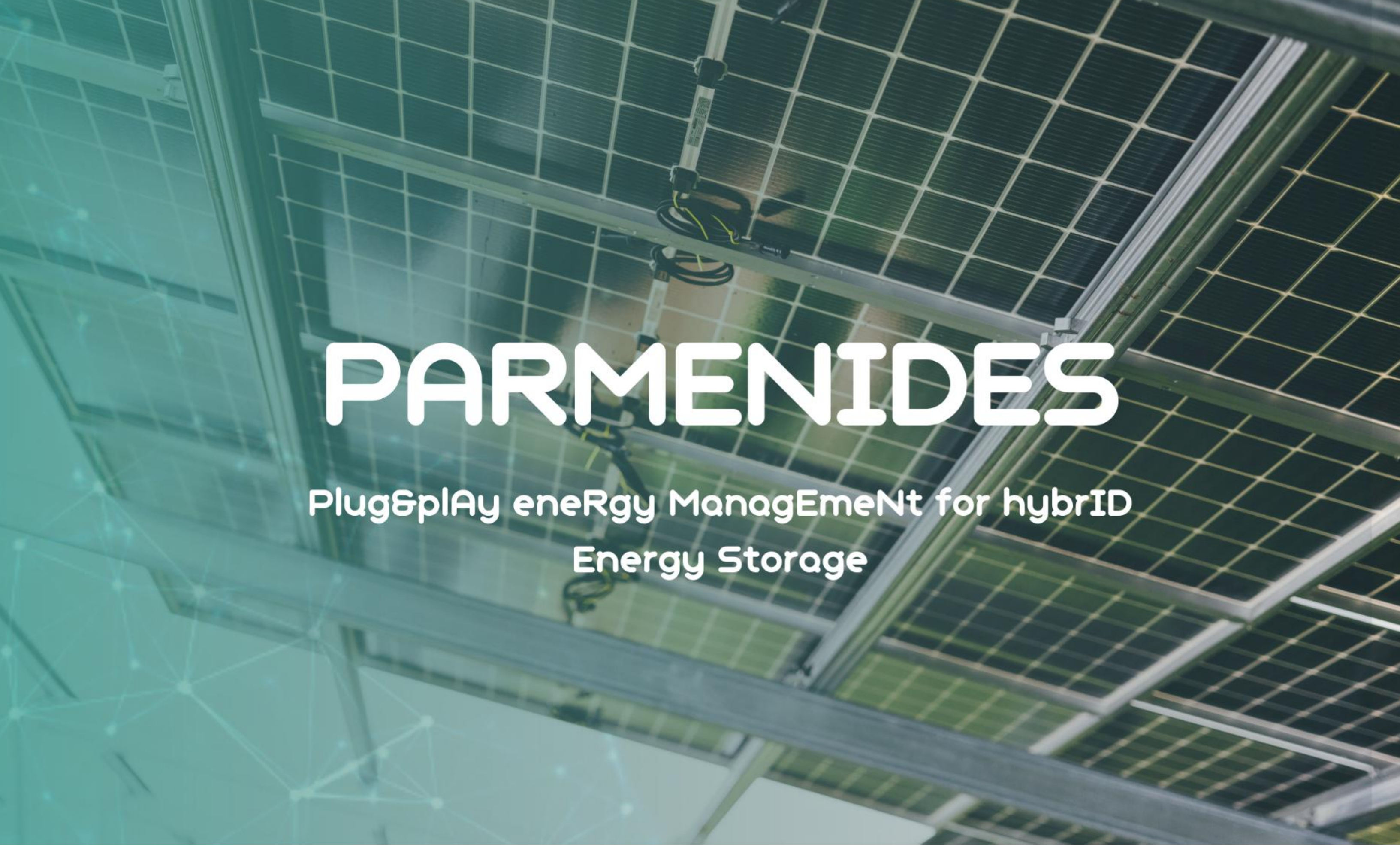


Deliverable D5.3

Evaluation and Validation Report

Work Package 5

Disclaimer

Funded by the European Union's Horizon Europe programme under Grant Agreement nº 101096453

Grant agreement	101096453
Type of action	HORIZON-IA HORIZON Innovation Actions
Topic	HORIZON-CL5-2022-D3-01-10 Interoperable solutions for flexibility services using distributed energy storage
Starting date of project	01.01.2023
Project duration	36 months

Work package	WP5 – PARMENIDES system evaluation
Related task	T5.3 – Demonstration Austrian pilot T5.4 – Demonstration Swedish pilot T5.5 – Results, replicability, and scalability analysis
Deliverable due date	M36 (31.12.2025)
Actual delivery date	M38 (20.02.2026)
Dissemination level	PU – Public
Deliverable responsible	KTH

Document Information

Document Version: 1.0

Revision / Status: Submission

All Authors/Partners

Name	Organisation
Lorenz Ray Payonga	KTH Royal Institute of Technology
Hatef Madani	KTH Royal Institute of Technology
Davide Rolando	KTH Royal Institute of Technology
Jörgen Wallin	KTH Royal Institute of Technology
Mark Stefan	AIT Austrian Institute of Technology
Marc Dünser	AIT Austrian Institute of Technology
Clemens Korner	AIT Austrian Institute of Technology
Jawad Kazmi	AIT Austrian Institute of Technology
Miloš Šipetić	AIT Austrian Institute of Technology
Vieri Emiliani	MAPS
Maria Aigner	ENS Energienetze Steiermark GmbH

Document History

Revision	Content/changes	Resp. partner	Date
0.1	Initial version & and structure	KTH	15.12.2025
0.2	Draft for review	KTH, AIT, MAPS	11.02.2025
0.3	First review	DERlab	11.02.2025
0.4	Implementation of revisions	KTH, AIT, MAPS	17.02.2025
0.5	Second review	DERlab	18.02.2025
1.0	Final version		20.02.2025

Document Approval

Final approval	Name	Resp. partner	Date
1.0	Mark Stefan	AIT	20.02.2026

Executive Summary

This deliverable reports on the evaluation and validation of the PARMENIDES system conducted within Work Package 5, integrating results from pilot demonstrations in Austria and Sweden together with a dedicated replicability and scalability analysis. The primary objective of WP5 is to assess the technical performance, interoperability, and flexibility potential of the developed solutions under real-world and controlled conditions

The Austrian pilot validated an ontology-driven, grid-conscious orchestration of Hybrid Energy Storage Systems (HESS) in two low-voltage distribution networks in Styria. By combining the PARMENIDES Energy Community Ontology (PECO), the Energy Management System for HESS (EMS4HESS), and Grid Capacity Management (GCM), the pilot demonstrated dynamic flexibility management under grid constraints. Results showed substantial reductions in grid imports and exports, improved self-consumption and self-sufficiency, and stable grid operation without physical reinforcement, confirming the feasibility of semantic, constraint-based control in energy communities

The Swedish pilot focused on demonstrating the PARMENIDES Flexibility Strategy and developing a HESS hardware-in-the-loop testbench at the KTH Granryd Laboratory. A staged simulation approach was used to train a reinforcement learning–based control agent, progressively increasing model fidelity and system complexity. The physical testbench provided experimental validation of storage behavior, control logic, and demand emulation using real data from the KTH Live-in Lab. An assessment of flexibility provision with HESS and its trade-offs between comfort, cost, and emissions was also performed.

In parallel, a replicability and scalability analysis was performed using virtual environments and scenario-based assessments. This analysis examined how the PARMENIDES architecture, control concepts, and ontology-based interoperability can be transferred to different community sizes, configurations, and operating conditions. Controller-in-the-loop testing and scenario evaluations were used to identify system limitations, scaling potential, and requirements for broader deployment. Together, the pilot results and analytical findings support the robustness, transferability, and future applicability of the PARMENIDES approach across diverse energy community contexts.

Table of contents

Abbreviations

Acronym	Description
AC	Alternating Current
AI	Artificial Intelligence
AIT	Austrian Institute of Technology
ANN	Artificial Neural Network
API	Application Programming Interface
BESS	Battery Energy Storage System
BMS	Building Management System
CIL	Controller-in-the-Loop
CO_2	Carbon Dioxide
COP	Coefficient of Performance
CPU	Central Processing Unit
DC	Direct Current
DHW	Domestic Hot Water
DNN	Deep Neural Network
DSO	Distribution System Operator
DSSE	Distribution System State Estimation
EC	Energy Community
EMS4HESS	Energy Management System for Hybrid Energy Storage Systems
ENS	Energienetze Steiermark
EV	Electric Vehicle
EVSE	Electric Vehicle Supply Equipment
FMU	Functional Mock-up Unit
FRP	Flexibility Requesting Party
GCM	Grid Capacity Management
GDPR	General Data Protection Regulation
HESS	Hybrid Energy Storage Systems
HIL	Hardware-in-the-Loop
HLUC	High-Level Use Case
HTTP	Hypertext Transfer Protocol
ICS	Information and Configuration System
ICT	Information and Communication Technology
IP	Internet Protocol

Acronym	Description
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
KTH	Kungliga Tekniska Högskolan (Royal Institute of Technology)
LHTES	Latent Heat Thermal Energy Storage
LV	Low Voltage
ML	Machine Learning
MLP	Multi-layer Perceptron
MQTT	Message Queuing Telemetry Transport
NIS2	Network and Information Security Directive (EU Directive 2022/2555)
ONNX	Open Neural Network Exchange
p.u.	per unit
PAC	Power Analysis and Control
PCM	Phase Change Material
PECO	PARMENIDES Energy Community Ontology
PID	Proportional-Integral-Derivative
PLC	Programmable Logic Controller
POD	Point of Delivery
PPO	Proximal Policy Optimization
PV	Photovoltaic
RBAC	Role-Based Access Control
RDP	Rapid Deployment Platform
REC	Renewable Energy Community
RED	Renewable Energy Directive
REST	Representational State Transfer
RL	Reinforcement Learning
RTU	Remote Terminal Unit
SAREF	Smart Applications Reference
SAREF4BLDG	SAREF extension for building devices
SHACL	Shapes Constraint Language
SHTES	Sensible Heat Thermal Energy Storage
SoC	State of Charge
SRA	Scalability and Replicability Analysis
TCP	Transmission Control Protocol

Acronym	Description
TES	Thermal Energy Storage
UC	Use Case
UFTP	USEF Flexibility Trading Protocol
USEF	Universal Smart Energy Framework
VLab	Virtual Laboratory
WAN	Wide Area Network
WP	Work Package
YAML	YAML Ain't Markup Language

1 Introduction

1.1 Scope

Work package 5: PARMENIDES system evaluation: The overarching goal of WP5 is to assemble the building bricks developed in the previous work packages into a real-world implementation and to evaluate its performances. Two different pilots will carry out this testing: one in Austria and one in Sweden. The generated data will serve different purposes such as: Comparison with virtual environment, validation of technical choices, quantification of storage-induced flexibility at different timescales.

Task 5.3: Demonstration Austrian pilot: Task 5.3 will demonstrate the solution for selected use cases in the Austrian municipalities of Gasen and Heimschuh. The demonstration phase will cover one year to be able to cover all seasonal conditions and to show the impact of short-term and long-term storage systems on flexibility. The deployed system will be continuously monitored to detect and fix failures immediately and to ensure a stable demonstration. In parallel to the pilot demonstration, the virtual environment will be used to elaborate additional use cases and community configuration (as an extension to the field demonstration). ENS is coordinating the demonstration phase and is responsible for regular interaction with the pilot customers, AIT will support the deployment and demonstration phase. Siemens Austria has equipped both communities in previous research projects. Therefore, a subcontract for Siemens Austria is planned at the budget of AIT, aiming to provide access to the field devices for AIT and to support the operation of the system. The defined KPIs (Task 2.2) will be evaluated. Reports (as part of D5.3) will be compiled based on all results and insights of the demonstration; a public report will be provided to the pilot customers.

Task 5.4: Demonstration Swedish pilot: Task 5.4 will demonstrate the second pilot in a different environment: the KTH Live-in-Lab. This pilot will run for a year, which will allow the HESS system (comprising heat pump integrating transient thermo-mechanical, thermal storage, electrical batteries, EV charging station) to operate under the full diversity of Northern European climate. Live-in-Lab capabilities to provide flexibility will be exploited to their full extent by establishing a dynamic interaction between end-users (and their data) and the pilot's EMS (deployed in Task 5.1). The building-related data generated by the facility will be used to assemble high-quality datasets. The EMS will be able to control each asset based on the use case scenarios (developed in Task 4.4). In particular, the pilot is expected to demonstrate the feasibility of providing very short-term flexibility (<1min) through heat pump thermo-mechanical process, short term flexibility through heat pumps speed control, battery systems (both electrical and thermal batteries), and seasonal flexibility through ground borehole heat exchangers. All these storage technologies are interconnected through the EMS and BMS on the lower layer. The defined KPIs (Task 2.2) will be evaluated. A pilot report (as part of D5.3) will be provided and distributed to a wide audience through Live-in-Lab communication channels, all results will be harmonized.

Task 5.5: Results, replicability, and scalability analysis: Task 5.5 runs in parallel to the demonstration of the pilots in Austria and Sweden and aims to analyse the results from the pilots. This includes technical, economic, and social aspects and is based on the defined use cases, KPIs, and requirements (WP2). Beside the activities in the pilots, comprehensive scalability and replicability analysis of the developed solutions will be performed within the virtual verification environment. Therefore, several scenarios with different configurations and sizes will be performed in order to identify gaps/potential to scale the solutions for bigger communities, other time resolutions, and other facilities, conditions, and constraints (technical, economic, social). All results will be harmonized and generalized to derive common recommendations for improving the system (ontology, architecture, EMS), future (research and development) activities, as well as for standardization activities (WP6). The defined KPIs (Task 2.2) will be evaluated. All results will be documented in D5.3 (together with the results from both pilots).

1.2 Structure of the document

This document is structured to present the evaluation and validation of the PARMENIDES system in a clear and systematic manner, covering pilot configurations, results, and cross-cutting analyses.

Section 1 introduces the scope and objectives of Work Package 5, outlining the role of the Austrian and Swedish pilots in evaluating the PARMENIDES system. It clarifies how the pilots, together with virtual assessments, contribute to validating technical choices, flexibility potential, and system performance.

Section 2 describes the configuration and implementation of the PARMENIDES pilots. It details the Austrian pilot deployments in Gasen and Heimschuh, including grid context, Hybrid Energy Storage Systems, communication architecture, and control concepts. The section also presents the Swedish pilot, covering the simulated HESS used to demonstrate the PARMENIDES Flexibility Strategy, the HESS Testbench at the KTH Granryd Laboratory, communication and control architecture, and the use of data from the KTH Live-in Lab.

Section 3 presents the results from the Austrian pilot. It focuses on dynamic grid-conscious flexibility management, the interaction between EMS4HESS and Grid Capacity Management, and the evaluation of pilot demonstrations. Key performance indicators are analysed to assess impacts on grid operation, self-consumption, and self-sufficiency.

Section 4 reports the results from the Swedish pilot. It summarizes outcomes from both the digital and physical infrastructures, including the demonstration of the PARMENIDES Flexibility Strategy using reinforcement learning, flexibility request generation, implementation of relevant use cases, and validation through hardware-in-the-loop testing on the HESS Testbench.

Section 5 addresses replicability and scalability. It presents the methodologies used to assess how the PARMENIDES solutions can be transferred and scaled to different energy community configurations, including virtual laboratory concepts, controller-in-the-loop testing, and scenario-based analyses.

Section 6 discusses challenges and opportunities identified across the Austrian pilot, Swedish pilot, EMS4HESS, and PECO, highlighting lessons learned and directions for future development.

2 Description and configuration of PARMENIDES pilots

This section provides an overview about the pilots in the PARMENIDES project.

2.1 Austrian Pilot

The Austrian pilot validated an ontology-driven, grid-aware orchestration of Hybrid Energy Storage Systems (HESS) in two low-voltage (LV) distribution areas in Styria – Gasen (AT1) and Heimschuh (AT2) – operated by Energienetze Steiermark (ENS). It demonstrated how dynamic, grid conscious limits coupled with a semantically interoperable energy management system (EMS4HESS) increased self-consumption, self-sufficiency, and grid-friendliness without physical reinforcement under the Austrian REC framework. The PECO ontology served as the common vocabulary across assets, actors, tariffs, schedules, and flexibility signals, ensuring end-to-end semantic alignment from field devices to optimization and user interfaces.

In line with Austria's transposition of RED II, the pilots focused on REC-like operations and provided grid-conscious flexibility under LV constraints (voltage rise, thermal loading, reverse power flow), emphasizing safe exploitation of storage and controllable loads as a community resource. The demonstrations thus combined semantic modelling (PECO), a distributed EMS (ROSE/EMS4HESS), and Grid Capacity Management (GCM) to deliver replicable, scalable flexibility at LV level.

2.1.1 Demonstration at Gasen

Grid and community context: Gasen was a rural LV grid with radial feeders and ≈ 350 kW contracted renewable capacity at the transformer. The participating community comprised 8 residential homes, 3 restaurants, one public building, and a district heating network. The HESS consisted of a community BESS (80 kW/140 kWh) and two EVSEs (1× public 22 kW, 1× private 22 kW). These resources were coordinated to absorb PV peaks, shift energy to evening demand, and respect dynamic LV constraints.

Instrumentation and observability: Each participant had a smart meter (15min), complemented by energyLIVE adapters providing 1min active/reactive power. Because energyLIVE did not provide voltage, Siemens PAC meters were deployed at selected customers and at the transformer (feeder and total) to capture per phase voltage/current and total power. A local weather station was installed to improve forecasting. Optimal voltage sensor placement was determined using ML-based state estimation, targeting ≈ 1 % of buses to balance accuracy and cost; this scheme enabled accurate voltage inference throughout the feeder without full instrumentation.

Communication and control: A site-local MQTT broker (industrial PC) published measurements and mirrored selected topics to a Siemens cloud broker. EMS4HESS (MAPS cloud) and AIT's GCM (RDP platform) subscribed to measurements, exchanged forecasts, and published limits/schedules/setpoints. A local RDP instance hosted the Clipper to enforce safe setpoints at 1min resolution and ensured fallback operation under WAN degradation. All streams and interfaces were aligned via PECO, ensuring semantic consistency across components.

The final deployment implemented the ML-guided voltage sensor placement, added the weather station, consolidated EMS4HESS as a ROSE Node Pair + Core for Austria, and updated the PECO instantiation to reflect final assets/tariffs and message semantics.

2.1.2 Demonstration at Heimschuh

Grid and community context: Heimschuh was a semiurban LV grid already at technical hosting limits due to high PV penetration (≈ 280 kW contracted renewables at the transformer). Participants included 5 households, 1 carpenter, and 1 oil mill. The HESS provided short-term flexibility via a BESS (100 kW/100 kWh); private EV charging was aligned with one selected customer to maximize local benefits under LV constraints.

Instrumentation and observability: Given the tight margins, the pilot was equipped with PAC devices at all relevant customers for 1min active/reactive power and voltage; DSSE complemented partial observability with ML-based voltage inference. The best performing sensor placements were decentralized and distant from the transformer, which improved observability at critical nodes while keeping measurement effort lean.

Communication and control: The architecture mirrored Gasen: an MQTT data fabric; GCM produced min/max power envelopes; EMS4HESS planned at 15min resolution; Clipper enforced real-time compliance at 1min cadence; and PECO ensured semantic interoperability.

2.1.3 Communication and Control

MQTT-centric data plane with semantic harmonization: The Austrian pilots employed a topic disciplined MQTT scheme for all field data and control – PAC voltages/currents/powers, energyLIVE active/reactive power, BESS SoC/DC/AC parameters, EVSE currents/energies, and the full set of GCM envelopes and EMS schedules. Topic structures and TimeValue/TimeSeries data models followed the pilot specification. PECO served as the semantic backbone, so that asset properties, PODs, ports, tariffs, user goals, messages, forecasts, and schedules were consistently modelled and discoverable by EMS and GCM. An overview is provided in Figure 1.

EMS4HESS on ROSE—distributed intelligence: EMS4HESS was deployed on the ROSE platform with a central ICS Knowledge Base (PECO compliant) and distributed nodes for local reasoning and resilience. A Decision Support System at ROSE Core coordinated AI forecasts, planning, and multi-objective optimization (e.g., cost vs. self-consumption), subject to GCM envelopes and tariff signals. Deployments were containerized and Kubernetes-orchestrated with GDPR and NIS2 controls (RBAC, encrypted comms, vulnerability management).

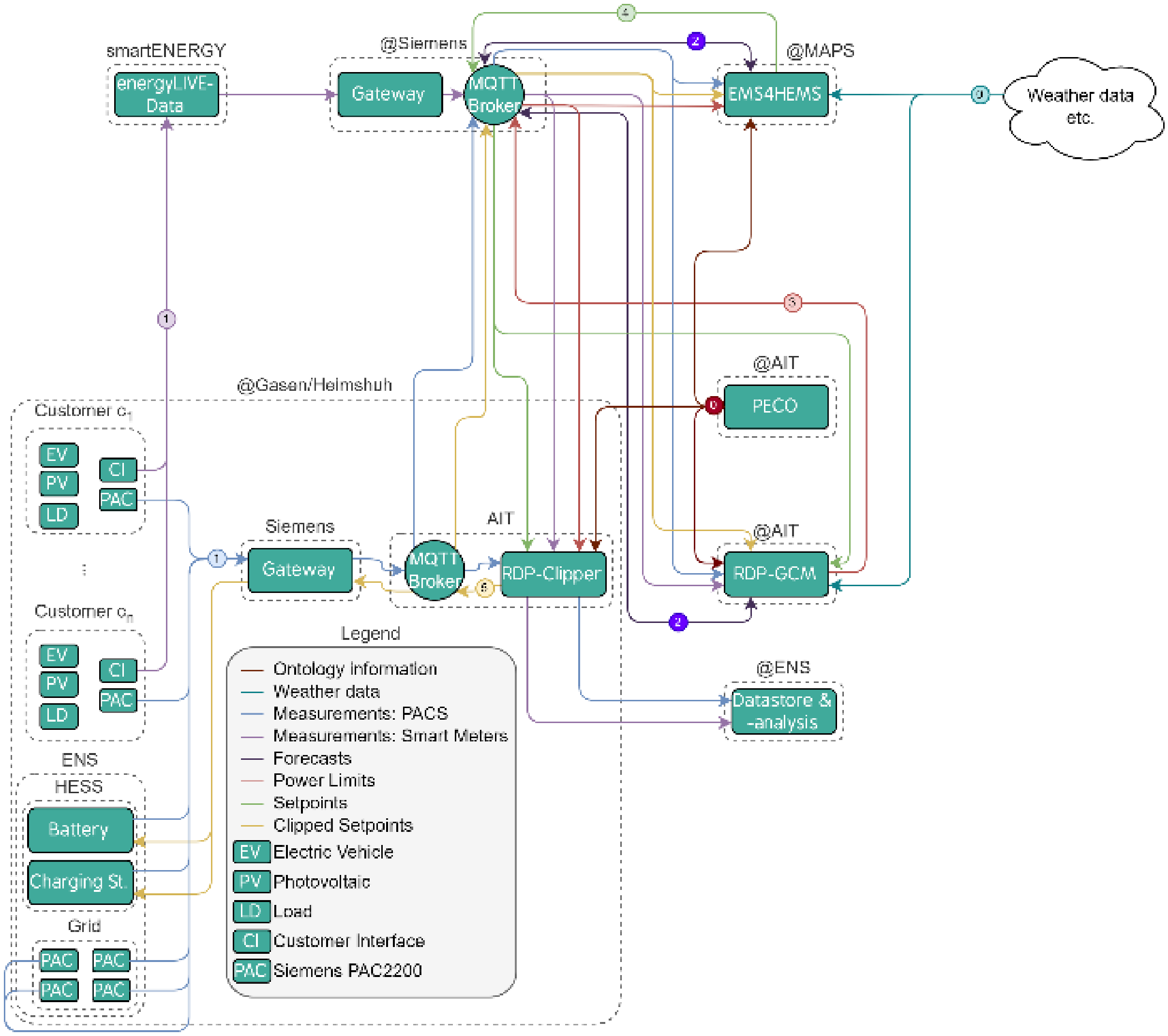


Figure 1: Architecture diagram of the Austrian pilots.

2.2 Swedish Pilot

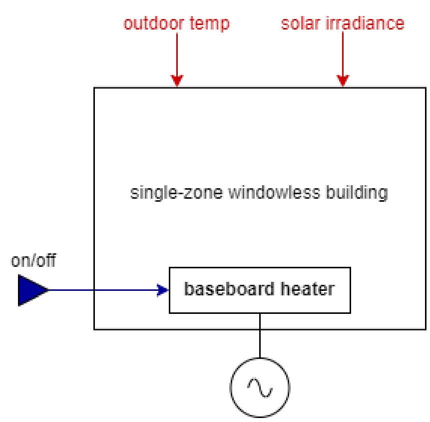
The Swedish Pilot activities have been narrowed to focus on two key aspects: (1) the demonstration of the PARMENIDES Flexibility Strategy and (2) the construction and operation of the HESS Testbench at the KTH Granryd Laboratory. The Flexibility Strategy has been demonstrated virtually, with the aim of approximating the characteristics of the HESS Testbench. Meanwhile, the HESS Testbench has been used to show the storage characteristics of different energy storage technologies, as well as to emulate the energy use of the KTH Live-in Lab.

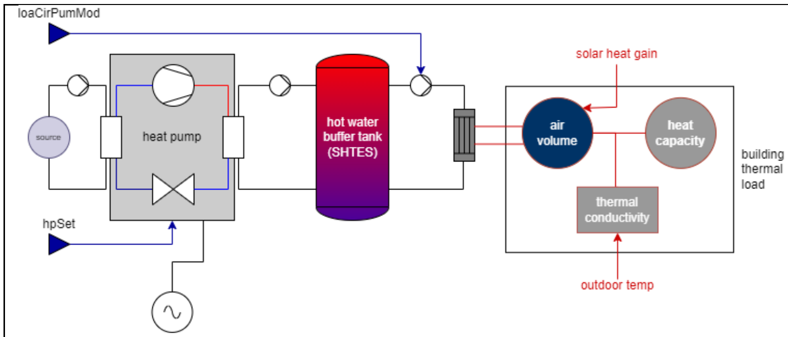
This section focuses on the description and configurations of the Swedish Pilot, while the tests and their results are detailed in Section 4.

2.2.1 Simulated HESS to demonstrate PARMENIDES Flexibility Strategy

The PARMENIDES Flexibility Strategy was developed, tested, and demonstrated in four stages, such that the complexity of the system increases at every iteration. The stages are described in Table 1 below.

Table 1. Evolution of simulations to demonstrate the PARMENIDES Flexibility Strategy.

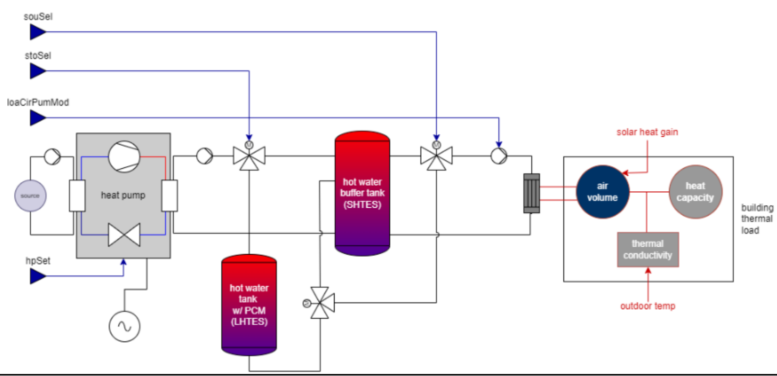
Name/Illustration	Description
Model 0 **SHTES**: Building thermal mass as energy storage (SHTES) 	Simple model where a residential building is directly heated by a baseboard heater without any buffer tank and any other form of energy storage. Building was represented as a hollow single-zone windowless enclosed space. The main objective is to test and demonstrate the trade-off exploration/exploitation approach (reinforcement learning) and establish the machine learning environment and architecture. Control was limited to activating and deactivating the heater. EnergyPlus was used as the building co-simulation software.
Model 1 **SHTES**: Building thermal mass + hot water buffer tank	A residential building is heated with a heat pump by a hot water buffer tank to allow more flexibility. Building was more accurately represented with calculations of solar heat gain that included windows. The water tank is an

<table>
<tr>
<td>


</td>
<td>approximation of the hot water buffer tank in the HESS Testbench at the Granryd Lab.

The main objective is to further improve the PARMENIDES Flexibility Strategy under more realistic physical conditions.

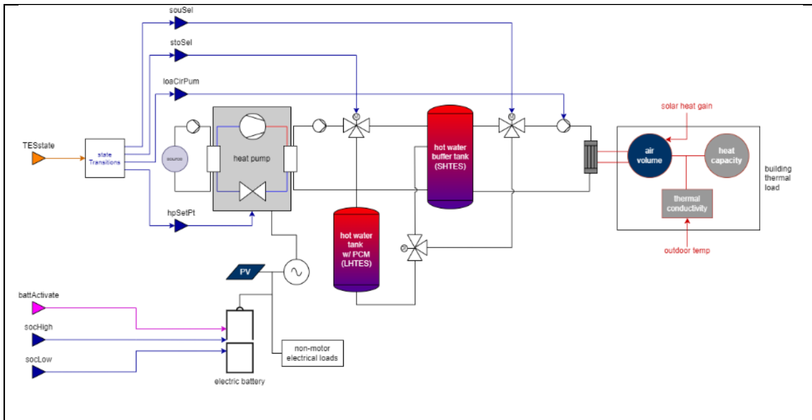
Heat pump set point and load-side circulation pump modulation can be controlled by the agent.

Dymola (Modelica) was used to model the system, but a functional mock-up unit (FMU) generated from it was used for co-simulation.</td>
</tr>
<tr>
<td>
Model 2
SHTES + LHTES: Building thermal mass + hot water buffer tank + PCM tank


</td>
<td>Extension of Model 1 with the addition of the PCM tank. The PCM tank was modelled according to the specifications of the PCM modules (Axiotherm ATS 44) and the vessel in the HESS Testbench.

The agent can choose among 1) charge SHTES, 2) discharge SHTES, 3) charge LHTES, 4) discharge LHTES, and 5) "do nothing" with pre-set configurations.</td>
</tr>
<tr>
<td>*Model 3*
SHTES + LHTES + Battery: Building thermal mass + hot water buffer tank + PCM tank + battery</td>
<td>Extension of Model 2 with the addition of the home battery and solar panels. The battery was modelled according to the specifications and test data of the battery in the HESS Testbench (EcoFlow DeltaPro).

Parameters of the hot water buffer tank, PCM tank, and valves have been updated based on physical tests.</td>
</tr>
</table>

	Non-motor electrical loads use the load profile of 50 apartments in KTH Live-in Lab. The agent can choose among the options in Model 2, but the deterministic logic of state transitions is implemented within the Modelica FMU.

Details of the key components in the simulation models are described in Table 2 below.

Table 2. Description of key components in simulation models.

Component (Relevant model(s))	Description	Notes
Residential building (0-3)	Conditioned floor area: 106.2 m^2 Air volume: 265.5 m^3 U-value (average): 0.238 W/m^2K Internal heat capacity: 45 Wh/m^2K	Variable solar heat gain modelled based on window specifications and solar irradiance data; Swedish archetype building from TABULA webtool[1]
Convective electric baseboard heater (0)	Power rating: 2000 W	Undersized to observe immediate effects of control; discarded in next iterations
Heat pump (1-3)	Heating capacity: 12 kW Refrigerant: R410A Compressor: Variable speed scroll-type COP: 4.9	Internal circulation pumps for evaporator and condenser were also modelled to approximate those present in the Thermia heat pump at the laboratory. Internal logic for pump and compressor modulation was adapted in Model 3 based on HESS testbench results.
Radiator (1-3)	Nominal T_{supply} / T_{return} = 45°C / 35°C	
Hot water buffer tank (1-3)	Volume: 0.75 m^3 Height: 0.12 m; Insulation: 0.12 m	Modelled as a stratified tank with 4 segments
PCM tank (2-3)	Volume: 0.293 m^3 (0.382 m^3 water – 0.089 m^3 PCM) Height: 0.75 m; Insulation: 0.12 m	The PCM capsules were simulated as one slab; due to the limitations of the Modelica PCM model,

[1] https://webtool.building-typology.eu/?c=se#bm

	Mass per capsule: 0.532 kg Number of capsules: 233 Latent heat: 230 kJ/kg Specific heat capacity: 3 kJ/kgK Density: 1.4 kg/L Heat conductivity: 0.6 W/mK Start of solidus: 43°C End of liquidus: 45°C	separate melting and congealing temperature ranges could not be modelled; start of solidus and end of solidus were used instead to approximate the characteristics.
Battery (3)	Voltage: 230 V Charging efficiency: 0.85 Discharging efficiency: 0.95 Maximum available charge: 3.6 kWh Maximum power: 3 kW	Internal logic was modelled based on the observed operation of the EcoFlow DeltaPro battery, particularly the stepped reduction in power draw at the last stage of charging, as well as how it manages charging and discharge within set upper and lower SOC. Self-discharge was not modelled.
Circulation pumps (1-3)	Nominal characteristics $m_{flow(load)}$: 0.98 kg/s; dp_{load}: 546.9 kPa $m_{flow(cond)}$: 0.75 kg/s; dp_{cond}: 37.4 kPa $m_{flow(evap)}$: 1.11 kg/s; dp_{evap}: 83.2 kPa	Load circulation pump was set to a fixed 0.15 modulation to attain ~1.7 m^3/h flow rate
Three-way valves (2-3)	Stroke time: 20 seconds	An additional 3-way valve, which is not present in the HESS testbench, was included in the junction connecting the two tanks and the load loop
PV panels (3)	PV 1: Area: 3.3 m^2; η: 0.192 PV 2: Area: 7.79 m^2; η: 0.20	

Apart from the components detailed above, heat power meters are also added in the same locations as in the HESS testbench. Pipes are also modelled to resemble the lengths and diameters of the pipes in the testbench for added realism. The evaporator loop, however, was not modelled according to geothermal borehole conditions.

The full HESS model implemented in Modelica is shown in Figure 2 below.

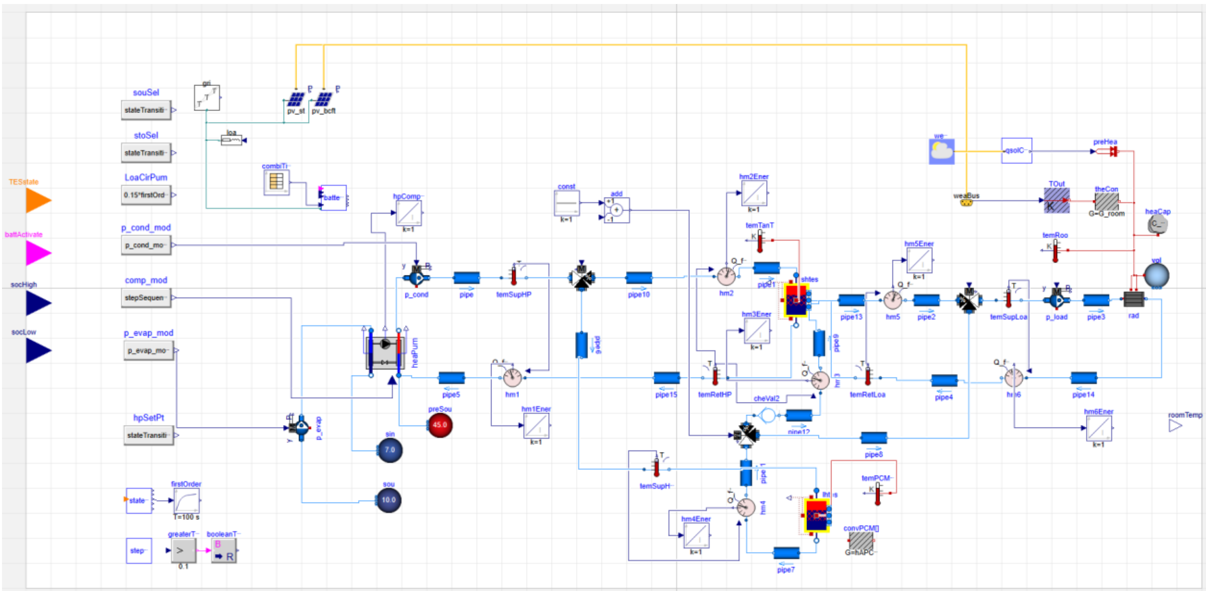


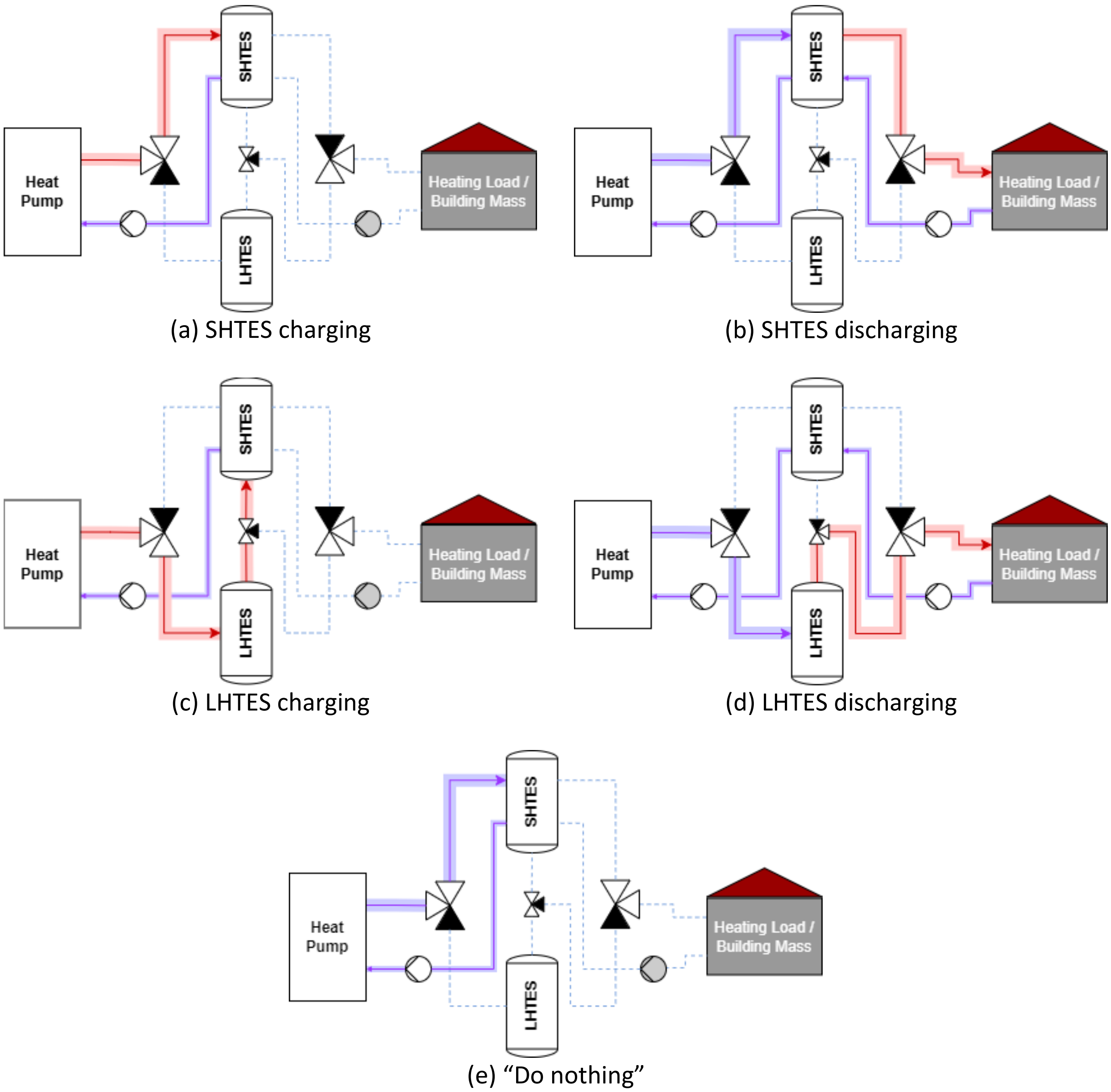
Figure 2. Full simulated HESS model for demonstration of PARMENIDES Flexibility Strategy

Considering the need to properly synchronize different components in the system when having multiple energy storage technologies, subjecting the heat pump and valves to trade-off exploration in RL as well as independent control was deemed risky. Therefore, in Models 2 and 3, only action options with safe pre-determined synchronized settings are allowed in training and in operation. These are described in Table 3 below, with flows illustrated in Figure 3.

Table 3. Action options for HESS operation (Models 2 and 3).

Action	Storage Selector Valve	Storage Source Valve	Load Circulation Pump	Heat Pump Setpoint (°C)
Charge SHTES	SHTES	LHTES[2]	OFF (0)	45
Discharge SHTES	SHTES	SHTES	ON (1)	30
Charge LHTES	LHTES	LHTES[3]	OFF (0)	55
Discharge LHTES	LHTES	LHTES	ON (1)	30
"Do nothing"	SHTES	LHTES[1]	OFF (0)	20

[2] To restrict circulation of water from SHTES
[3] Return flow guided by intermediate three-way valve and circulation in the heat pump loop



(a) SHTES charging

(b) SHTES discharging

(c) LHTES charging

(d) LHTES discharging

(e) "Do nothing"

Figure 3. Flow directions according to HESS agent action options for the thermal energy storage components

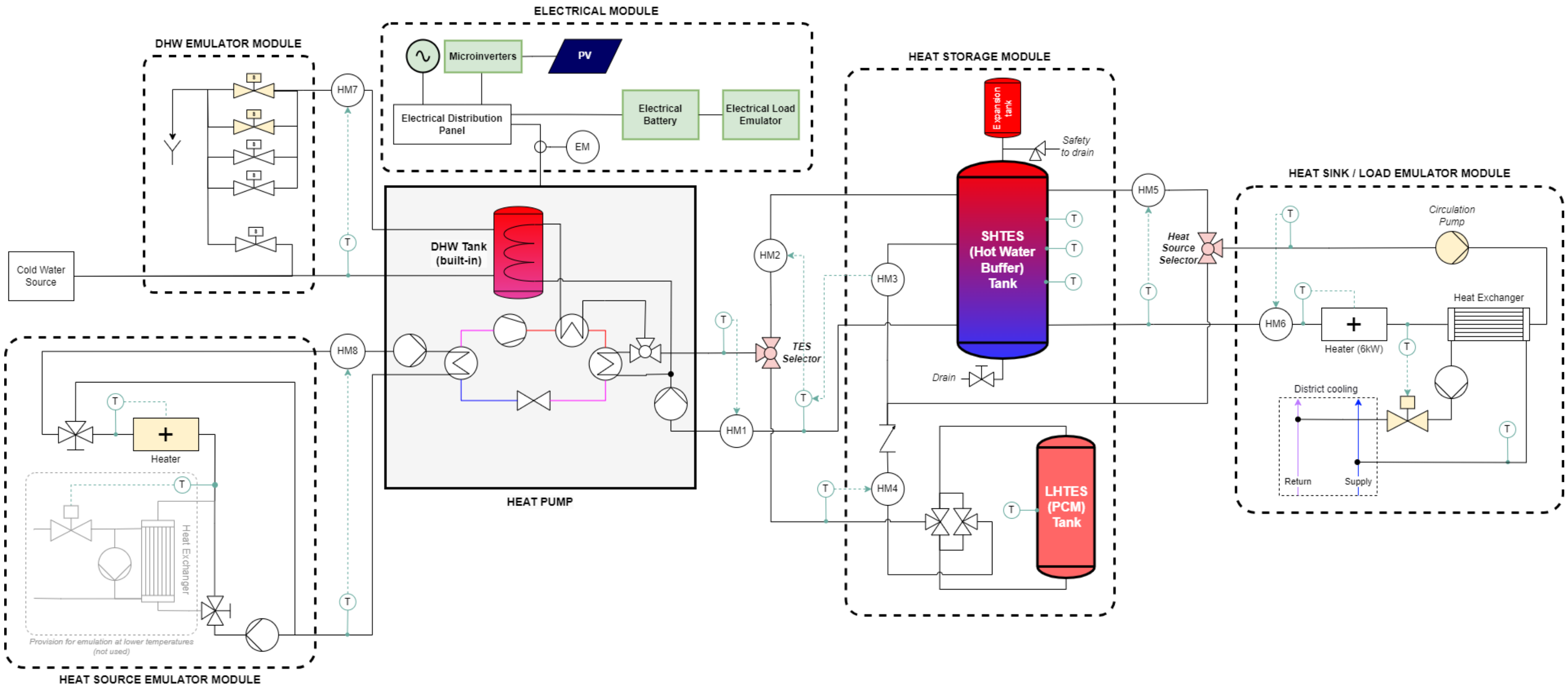


Figure 4. Schematic diagram of the HESS Testbench at KTH Granryd Laboratory

2.2.2 HESS Testbench at KTH Granryd Laboratory

The HESS Testbench at the KTH Granryd Laboratory is composed of five (5) modules in addition to a heat pump that couples electrical and thermal subsystems. These modules, shown in Figure 4 above, include a heat source emulator for the heat pump's evaporator loop, a heat sink/load emulator, a domestic hot water demand emulator, a heat storage module, and an electrical module. Controllable 3-way valves that correspond to those in the HESS simulations are shaded in light red. Components controlled for emulation are shaded in yellow. The modules are further described in Table 4. Note that the contents of the testbench closely resemble those of the HESS simulation because of the intention to use a controller trained with the PARMENIDES Flexibility Strategy on the testbench.

Table 4. Description of component modules of the HESS Testbench.

Module	Description
Heat pump	3-12 kW variable speed ground-source heat pump with 4.6 kW compressor power, R410a refrigerant, polypropylene glycol + water source loop heat transfer fluid, COP 4.6, 20-65°C heating temperature, built-in 180L DHW tank
Heat source emulator	Emulator suitable for emulating a broad range of ground source temperatures, containing both a heat and cold source. In PARMENIDES, the heater is modulated for this purpose.
Heat sink/load emulator	Emulator capable of emulating heat demand with a heat rejection component connected to the district cooling network. This is done by modulating a regulation valve.
Domestic hot water demand emulator	Emulator capable of emulating domestic hot water demand by controlling motorized valves.
Heat storage module	Comprised of a 741-L hot water buffer tank and a 382-L water tank containing 89 L of inorganic PCM with melting temperature of 45-48°C and 41-44°C congealing temperature.
Electrical module	Contains PV modules AC-coupled with microinverters, a 3.6 kWh battery, and non-motor electrical loads. In PARMENIDES, a fixed 830W load was used to test battery charging and discharging profiles.

The HESS Testbench was used to observe the charging and discharging profiles of the component energy storage technologies, note practical constraints and considerations in improving the simulation models, and emulate the demand from the KTH Live-in Lab. The testbench can be operated manually but can also be set to operate automatically given a demand profile, as shown in Figure 5.

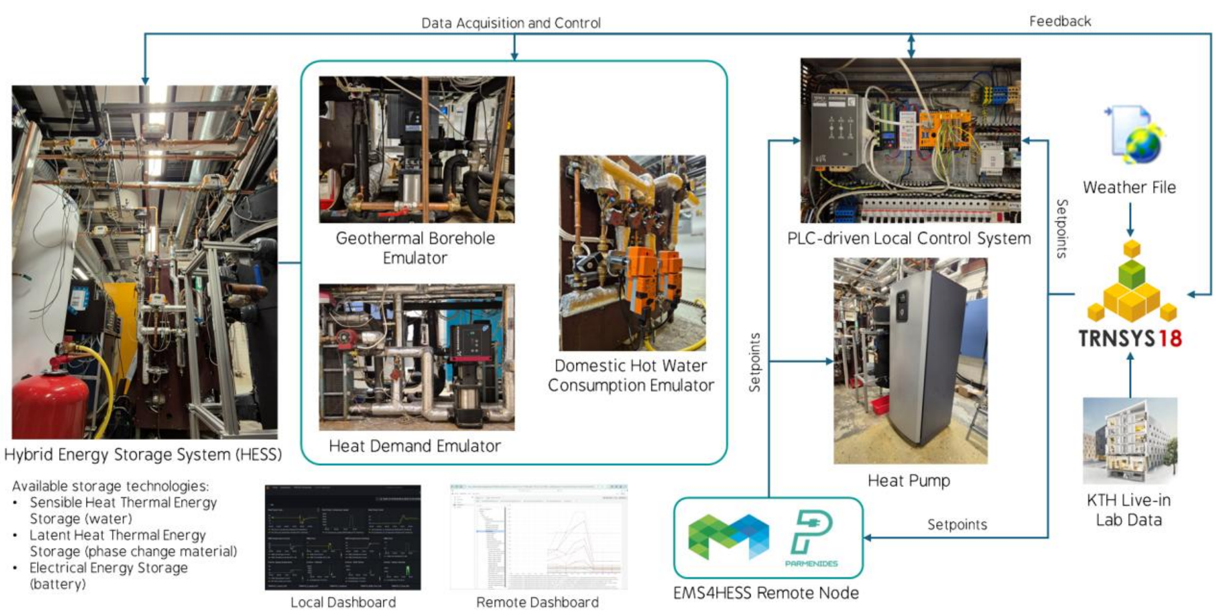


Figure 5. HESS Testbench operation architecture

Each of the three emulators, i.e., heat source (in this use case, a geothermal borehole), heat demand, and domestic hot water consumption, can be emulated based on simulations or demand profiles. In the case of the KTH Live-in Lab demand emulation, the ground source borehole temperatures were emulated based on a weather file, while domestic hot water and space heating demands were based on historical load profiles.

The manual operation for charging and discharging tests follow the configurations in Table 3.

2.2.3 Communication and control

The monitoring and control of the HESS Testbench are implemented using a PLC-like industrial control system based on a Revolution Pi 4 (RevPi). The RevPi is operated as a *soft PLC*, running a Linux operating system and Python-based control services, while interfacing directly with industrial I/O modules and field devices. This architecture enables flexible implementation of advanced control logic while maintaining compatibility with standard industrial communication protocols.

Control hardware and I/O architecture

The RevPi setup consists of multiple expansion modules to interface with the physical components of the testbench:

- A RevPi Connect 4 module acts as the main controller and communication gateway.
- A RevPi MIO module is used for digital inputs and outputs.
- A RevPi AIO module is used for analog inputs and outputs (0–10 V).
- A RevPi RO module provides relay outputs for switching operations.

Motorized valves, relay-controlled valves, and electrical heater commands are interfaced through a combination of digital and analog I/O. Analog outputs are used in particular for continuously controlled actuators, such as three-way valves and heater power modulation.

Communication with the heat pump and instrumentation

The RevPi communicates with the Thermia heat pump using the Modbus TCP protocol via the RevPi Connect 4 module. The software implementation for this communication is based on the open-source Python library *pythermiagenesis*[4], which provides access to operational variables and setpoints of the heat pump.

The RevPi also collects measurement data from several field devices installed in the HESS Testbench:

- Eight 2Flow E-modus RC82 ultrasonic heat meters, connected via Modbus RTU over RS485, are used to measure thermal energy flows in the different hydraulic circuits.
- Temperature sensors installed in the storage tanks are read via Modbus TCP.

[4] https://github.com/cjne/pythermiagenesis

- Electrical power consumption is monitored using a Siemens SENTRON 7KM2200 electricity meter, also accessed via Modbus TCP.

This combination of Modbus RTU and Modbus TCP allows integration of both legacy field instrumentation and IP-based devices within a unified control architecture.

Software architecture and data management

A Python-based software stack running on the RevPi manages the overall monitoring, control, and communication tasks of the HESS Testbench. Two background monitoring services operate continuously, independent of Hardware-in-the-Loop (HIL) experiments:

- Heat pump monitoring service, which polls operational variables from the heat pump.
- Test rig monitoring service, which collects measurements from sensors and meters installed on the testbench.

These monitoring services run at a 30-second interval and ensure continuous data acquisition during both manual operation and automated experiments.

All collected measurements are:

- Published to the EMS4HESS MQTT broker for integration with higher-level services, and
- Stored in a local InfluxDB database hosted at the KTH Granryd Laboratory.

The same server hosting the InfluxDB instance also runs Grafana, which is used for real-time visualization and post-processing of the collected data.

Communication between TRNSYS and the HESS Testbench

The TRNSYS simulation environment runs on a dedicated computer at the KTH Granryd Laboratory. To ensure reliable and deterministic coupling between TRNSYS and the physical testbench, communication between TRNSYS and the RevPi is implemented using direct TCP/IP messaging rather than MQTT. All MQTT publishing is centralized on the RevPi.

TRNSYS is operated in real-time mode, meaning that each simulation time step corresponds directly to real elapsed time. This ensures consistent temporal alignment between the simulated system and the physical testbench. The TRNSYS simulation time step is 1 minute.

Hardware-in-the-Loop control logic

Closed-loop control during HIL experiments is implemented via a dedicated Python service on the RevPi, which operates at a 10-second interval. This service synchronizes simulation outputs with physical measurements and updates actuator commands accordingly.

Several PID controllers are implemented in software on the RevPi to align the physical behavior of the HESS Testbench with the TRNSYS simulation outputs. These include:

- Control of the return temperature from the load by modulating a motorized regulation valve via a 0-10 V analog output.
- Control of domestic hot water (DHW) flow by adjusting one or two DHW control valves.
- Control of the borehole heat exchanger (BHE) loop temperature by regulating the electrical heater power via a 0-5 V analog signal.

In each case, the controller compares locally measured values with the corresponding target values provided by TRNSYS and adjusts actuator commands accordingly. The real-time exchange of data between TRNSYS and the physical testbench, combined with software-based closed-loop control, constitutes a Hardware-in-the-Loop (HIL) simulation setup.

Operational considerations and safety

Manual override of actuators is possible, allowing operators to intervene during experiments. Software-based safety limits are implemented to protect the heat pump, for example by shutting it down if evaporator temperatures fall below predefined thresholds. In addition, the electrical heaters are equipped with independent hardware safety switches that disconnect power in case of overheating.

No automated watchdog or fallback control modes are currently implemented. In the event of communication loss between TRNSYS and the RevPi, the control system holds the last valid actuator commands. For this reason, the current setup is not intended for unattended long-term operation, but rather for supervised experimental use within the laboratory environment.

2.2.4 Data from KTH Live-in Lab

To represent realistic building demand conditions in the simulation and Hardware-in-the-Loop experiments, historical operational data from the KTH Live-In Lab were used as input to the TRNSYS model. The KTH Live-In Lab is a residential testbed consisting of 305 apartments, from which detailed measurements of thermal and domestic hot water usage are continuously collected.

For the purposes of this work, a subset of historical data corresponding to 50 apartments was selected. The data were originally collected from real operation of the residential buildings and therefore reflect realistic occupant behavior, domestic hot water usage patterns, and space heating demand variations over time.

Data structure and temporal resolution

The input data provided to TRNSYS are structured as time series with a one-minute temporal resolution. Each data record includes thermal and hydraulic variables representative of the heating and domestic hot water subsystems, including:

- Borehole or heat source temperature indicators,
- Supply temperatures,
- Mass flow rates for space heating and domestic hot water,
- Thermal power demand.

The one-minute resolution allows the capture of short-term demand fluctuations while remaining compatible with the real-time operation of the TRNSYS simulation and the HESS Testbench.

Data adaptation and scaling

The original Live-In Lab data were collected at a scale corresponding to residential buildings significantly larger than the HESS Testbench and its associated Thermia heat pump. Therefore, a scaling procedure was applied prior to using the data as input to TRNSYS.

This scaling was performed to ensure that thermal power levels, mass flow rates, and temperature ranges were compatible with the nominal operating range and capacity of the heat pump and hydraulic components installed in the HESS Testbench.

The scaling was carried out as a preprocessing step, with the objective of preserving the temporal dynamics and relative variability of the original demand profiles while adapting their absolute magnitude to the laboratory system. The focus of this work was on the behavior of the control and flexibility strategies rather than on reproducing absolute building-scale energy quantities.

Use within TRNSYS and HIL experiments

Within TRNSYS, the adapted Live-In Lab data are used as external boundary conditions that drive the simulated heating and domestic hot water demand. During Hardware-in-the-Loop operation, these demand signals are synchronized in real time with the physical HESS Testbench through the communication architecture described in the previous section.

As a result, the physical testbench is exposed to demand profiles that originate from real residential operation, while operating at a scale compatible with laboratory constraints. This approach allows testing of control strategies under realistic, time-varying conditions without requiring full-scale building hardware.

3 Results from the Austrian Pilot

3.1 Dynamic Grid-Conscious Flexibility Management

Integrating distributed energy resources into low-voltage grids creates challenges such as voltage instability, thermal over-load, and bidirectional power flows. Traditionally, operators address these issues using static injection limits or expensive infrastructure upgrades. The Austrian PARMENIDES pilot implementation proposes an alternative approach based on dynamic flexibility management, enabled by real-time forecasting, semantic modelling, and coordinated control.

At the core of this approach is the Grid Capacity Management (GCM) system, which operates as a constraint engine for the EMS4HESS. GCM periodically computes upper and lower bounds for active and reactive power for each controllable asset as part of the HESS. These bounds are derived from short-term forecasts of load and generation, combined with a detailed model of the distribution grid topology and asset characteristics. GCM is deployed using AIT's Rapid Deployment Platform (RDP), which is a technology stack that enables fast deployment of energy and capacity management systems in real world applications. It is built on a modular, containerised architecture, allowing rapid scaling and flexible deployment[5].

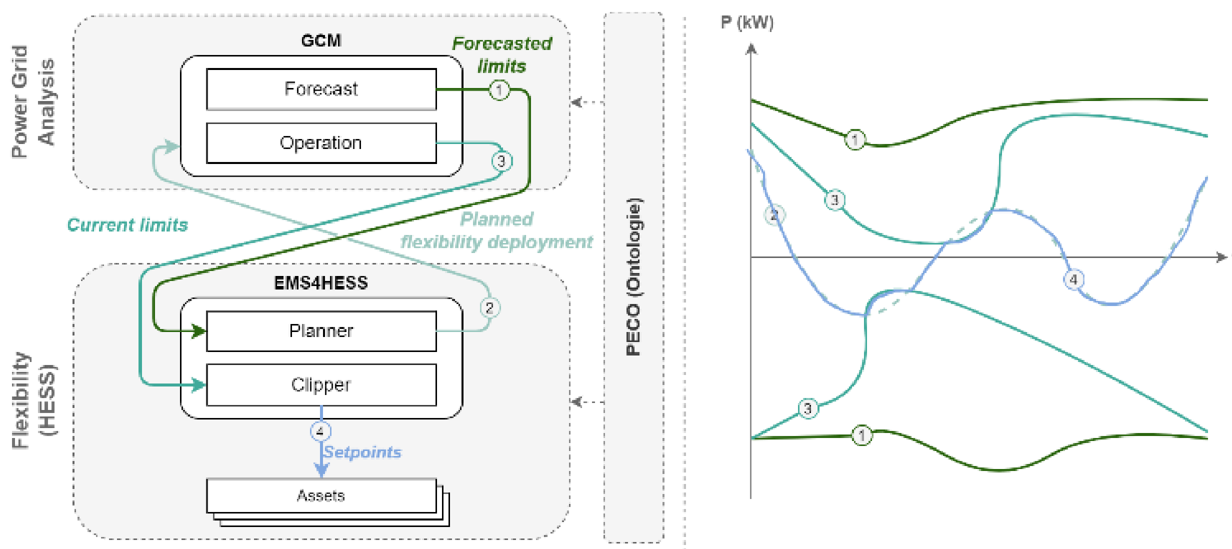


Figure 6: Data flow diagram of the interacting systems as part of the Austrian pilots.

At the core of the GCM is a module performing Distribution System State Estimation (DSSE) using machine learning models, which has been trained offline on synthetic load flow data. The estimation process leverages a minimal set of strategically placed voltage sensors, identified through a data-driven optimisation procedure. Artificial Neural Networks (ANNs) are trained to predict the unknown voltage values based on

---

[5] M. Spiegel, C. Korner, D. Vettoretti, J. Kathan and M. Stefan, "Rapid deployment of capacity and energy management systems," in Proc. IET Conf., 2024

these few known measurements. The usage of these ANNs enables accurate and very fast grid state reconstruction even in sparsely instrumented environments[6] [7].

The left side of Figure 6 shows, in a data flow diagram, an exemplary representation of how the deployed software and hardware components interact. On the right side, the figure illustrates, in a plot, the data exchanged for the HESS.

Three main systems are involved:

- GCM
- EMS4HESS
- PECO

A PECO instantiation of the pilots provides the necessary information about the energy community, their customers, and the available assets to the other two systems. This information is used to derive their respective system configurations and thus ensures semantic interoperability between the systems. Subsequently, GCM and EMS interact as follows:

1. **Forecasted limits:** Based on historical data, GCM periodically generates forecasts for the load and generation situation up to 24 hours ahead. Based on these forecasts, dynamic power limits for the HESS are calculated and transmitted to the EMS.
2. **Planned flexibility deployment:** Taking the forecasted power limits into account, the Planner module of the EMS4HESS optimises the use of flexibility, for instance to perform self-consumption optimisation. Based on this optimisation, the EMS4HESS transmits to GCM in real-time the currently planned flexibility deployment.
3. **Current limits:** Using the current measurements and the planned flexibility deployment, GCM calculates the present network state in real-time and determines the updated power limits. These limits are then sent to the EMS4HESS.
4. **Setpoints:** The EMS4HESS must ensure that during flexibility activation, the power levels remain within the current power limits determined by GCM. Therefore, the EMS4HESS uses a clipping algorithm, where the planned current flexibility deployment is clipped according to the current power limits.

This layered control architecture, combining predictive analytics, semantic modelling, and real-time constraint enforcement, enables the energy community to operate in a grid-supportive manner. It allows for dynamic adaptation to grid conditions, supports multiple flexibility strategies, and ensures safe integration of high shares of renewable energy without compromising grid reliability.

---

[6] M. Dünser, B.-V. Rao and S. Reisenbauer, "Ideal sensor placement and state estimation in distribution grids," in Proc. IET Conf., 2025

[7] D. Fellner, M. Stefan, B.-V. Rao, S. Reisenbauer, M. Aigner and G. Taljan, "PARMENIDES – ideal voltage sensory placement for battery storage operation optimization," in Proc. IET Conf., Vienna, Austria, 2024.

3.2 Grid Capacity Management (GCM)

3.2.1 Introduction

The Grid Capacity Management (GCM) framework developed at AIT provides a fast and lightweight approach for machine-learning-based state estimation and the derivation of operating envelopes for grid-connected flexibilities. In PARMENIDES, substantial enhancements have been made to both the technical backend and methodological components, driven largely by developments within the project as well as complementary national projects such as INFRADAPT and INNOnet.

The core idea of GCM is to estimate grid states using a low-latency machine-learning model and to use this estimation to compute operating envelopes for flexible resources. These envelopes are subsequently transferred to systems such as EMS4HESS for energy management.

3.2.2 Measurement Infrastructure and Model training

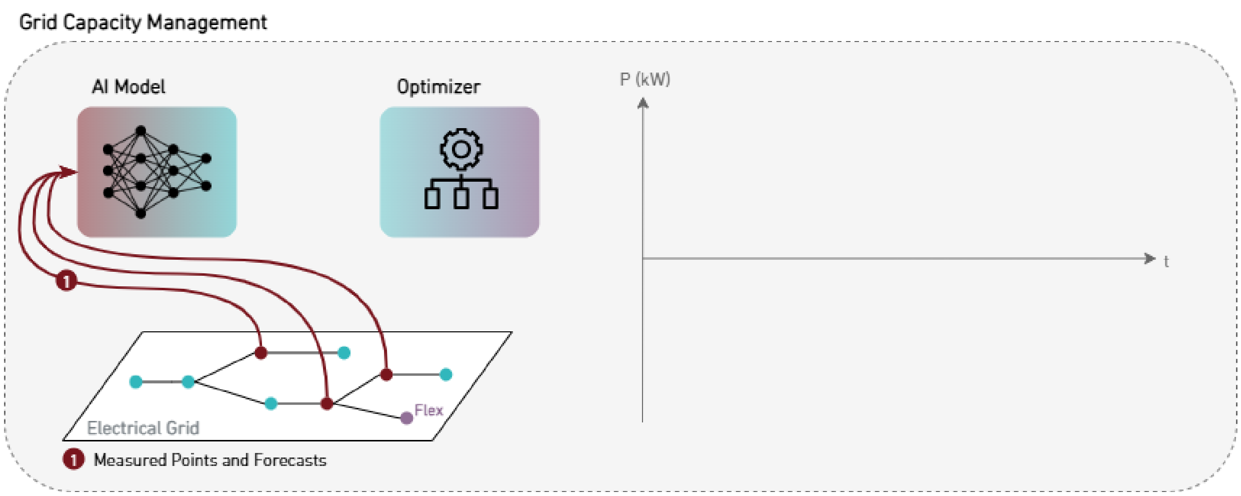


Figure 7: Measurement from pilots used for AI model training.

The Austrian pilot sites Gasen and Heimschuh provide a limited number of measurement points, and their spatial distribution is not ideal for detailed statistical assessments. Heimschuh includes seven sensors, which is generally sufficient, while Gasen includes only two to three voltage measurements, constraining interpretability.

Sensor placement in Heimschuh shows that not all feeders are measured and that some measurements are clustered (see Figure 8) along the same feeders.

To train the ML-based state estimation, AIT employs an internal python-based simulation framework built around the open-source power-flow calculation tool *pandapower*. This framework is capable of running large numbers of load-flow simulations for different operating scenarios, even for large and complex LV-grids. Monte-Carlo simulations, annual load profile approaches, or combinations thereof can be generated, according to the user’s needs. Parts of the functionality of this simulation framework were developed in the PARMENIDES project.

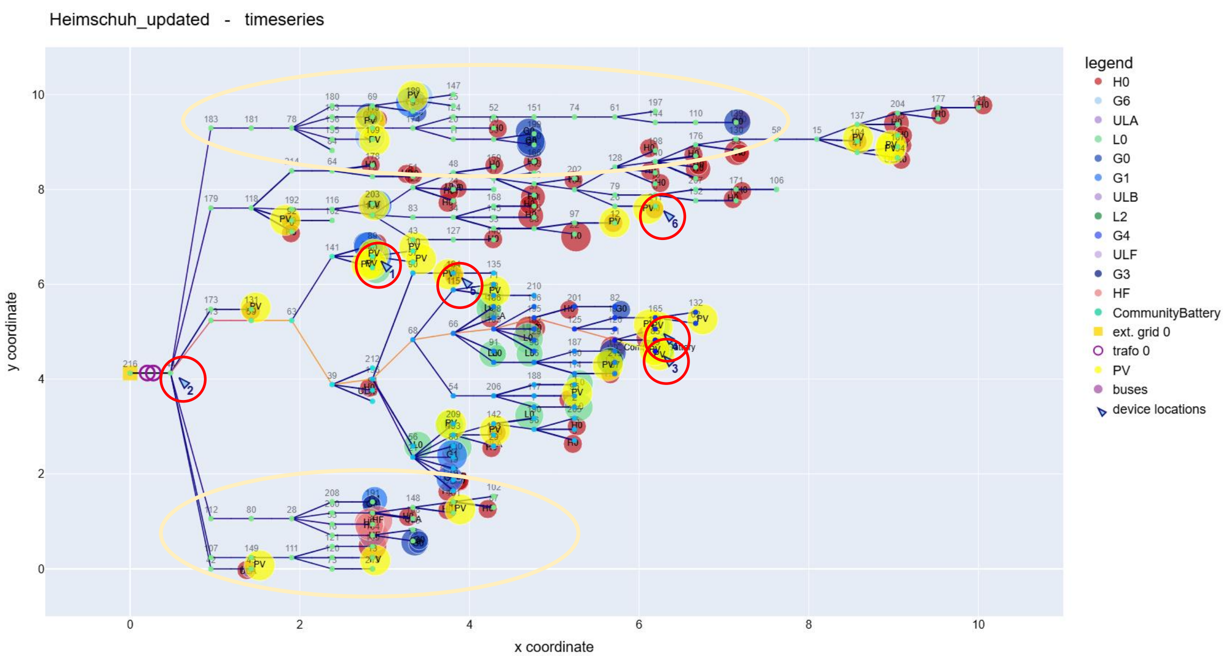


Figure 8: Schematic representation of the low voltage distribution grid in Heimschuh. The two encircled feeders are entirely unobserved.

3.2.3 Machine-Learning-Based State Estimation

The state estimation itself is performed using a fully connected multi-layer perceptron (MLP). Although the first implementation relied on TensorFlow for both training and inference, the inference phase has now been migrated to the ONNX format to improve inference time and size of the deployed module container. This change significantly enhances performance while retaining the simple but effective neural-network architecture, at no loss in accuracy. By this change, the measured inference time, including the derivation of the downstream setpoints and operating envelopes has been reduced by a factor of about 5-10, depending on the compute power used.

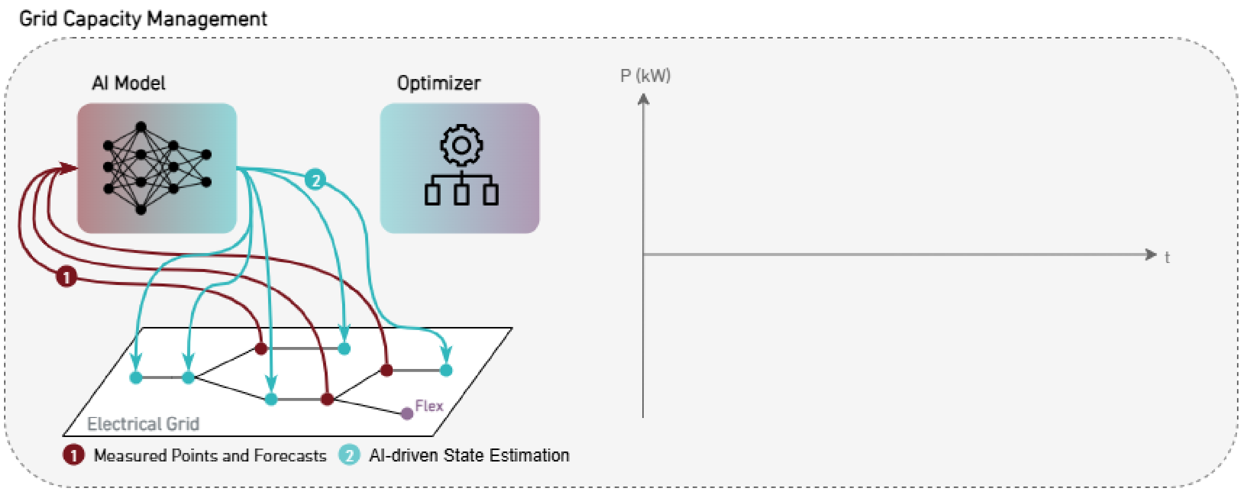


Figure 9: AI-driven state estimation provides voltages for all unmeasured points in the grid.

3.2.4 Optimisation Based on Statistical Relationships

The optimization step – one of the components that changed most within the course of PARMENIDES – uses covariance matrices to quantify how grid voltages respond to changes in flexibility setpoints. This method is computationally efficient, widely applicable to different grid parameters, and does not require ML inference during optimization. Therefore, the deployed version of the GCM does not necessitate any recalculation of load-flows, which would considerably slow down the entire process. However, it depends strongly on the quality of the simulated data and assumes approximately Gaussian behaviour of the observables in the covariance matrices, making second-order effects harder to capture in the calculation of the setpoints. However, it was tested and validated that the underlying variables approximately follow Gaussian distributions to ensure the validity of the covariance-based calculations.

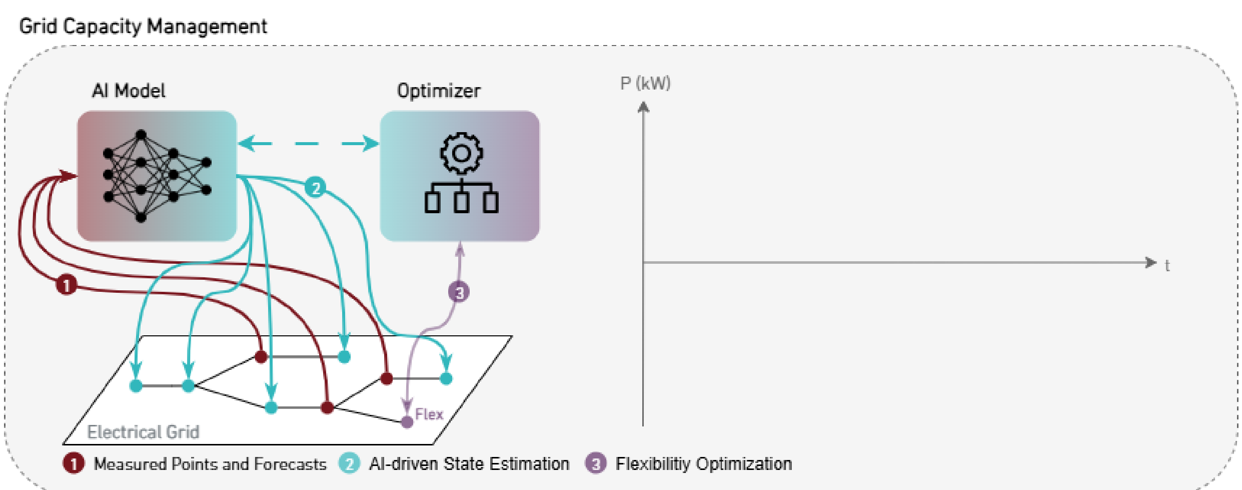


Figure 10: Optimisation based on statistical relationships between flexibility setpoints and grid voltages.

3.2.5 Deriving Operating Envelopes

Once the influence of flexibility setpoints on voltage levels has been quantified, GCM computes operating envelopes using a binary search approach ("half-interval search"). This technique reduces the search space by half at each iteration, enabling extremely fast envelope generation. The method also includes tuneable sensitivity parameters and initial max/min checks to improve performance and reduce latency.

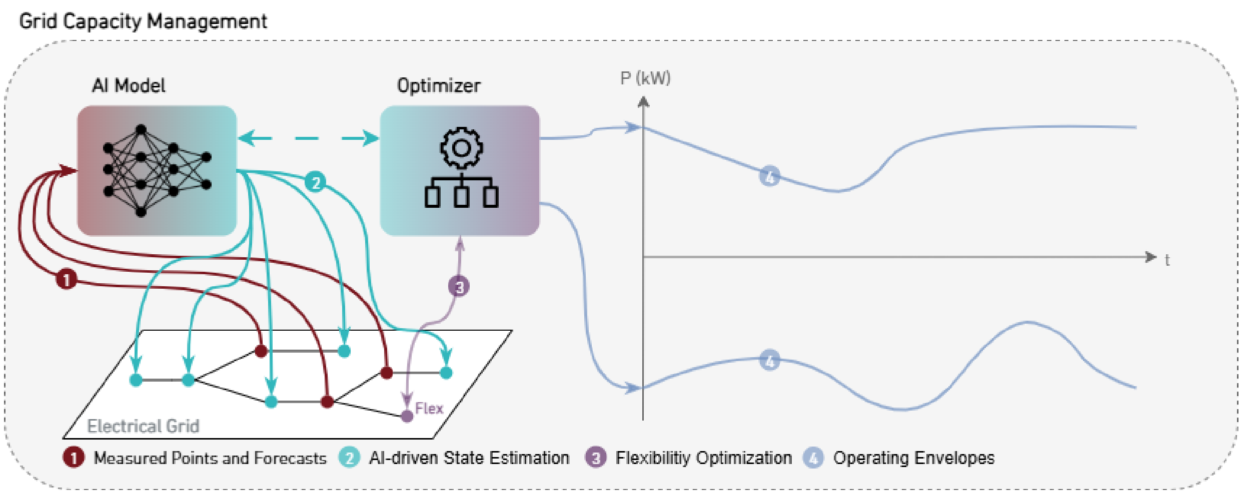


Figure 11: Calculated operational envelopes for flexible assets.

3.2.6 Computing the "Ideal" Setpoint

Beyond operating envelopes, GCM can determine a "grid-ideal" setpoint – i.e., a flexibility operating point that keeps voltages as close as possible to a target value, often 1.0 p.u. This optimization is mathematically more challenging and is implemented using a trust-region algorithm, which balances performance in terms of accuracy versus speed. Although this function is not central to PARMENIDES, it may be valuable for future applications.

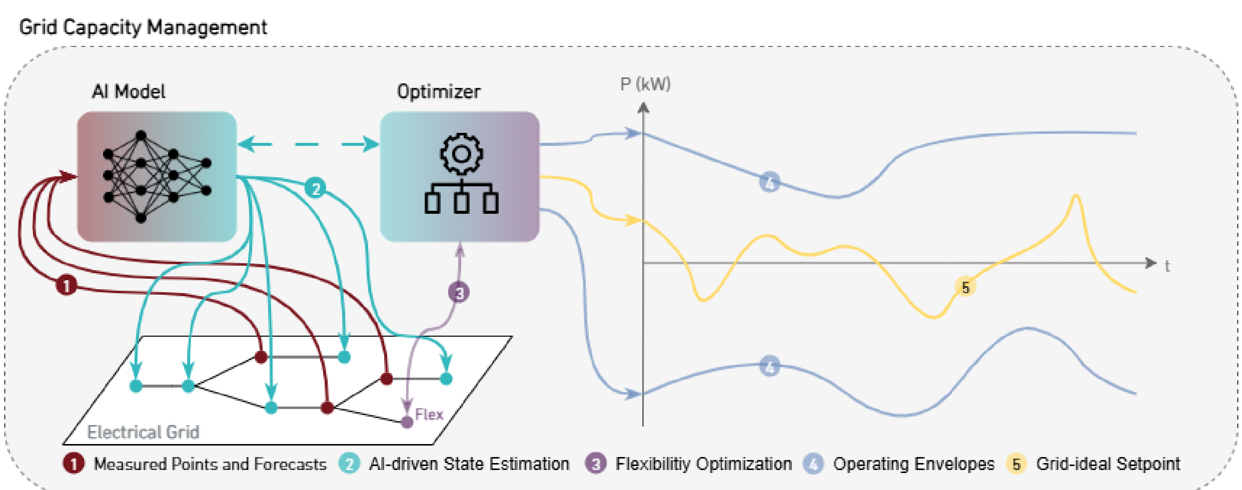


Figure 12: "Ideal" setpoint for flexibilities in the grid, allowing for operation to minimize voltage variations.

3.2.7 Performance Achievements

Efforts in PARMENIDES have focused on making GCM more lightweight and faster. The total software package has been reduced in size by more than 80 % compared to the initial version, now requiring well below 1 GB for the full deployment. Runtime performance has also improved significantly: the complete workflow – including ML state estimation, operating envelope computation, and ideal-setpoint calculation – runs in about 200 ms on a standard laptop.

3.3 Evaluation and Pilot Demonstration

Initial results demonstrate the effectiveness of the implemented architecture in maintaining grid stability and optimising energy flows. Voltage measurements across community members and transformer phases remained within acceptable limits, and the HESS successfully shifted energy from periods of high generation to periods of high consumption.

The coordinated operation of EMS4HESS and GCM enabled compliance with grid constraints, and the Clipper module ensured safe operation under dynamic conditions. Time-series data of charging and discharging profiles confirmed the system's ability to follow optimised schedules. These findings validate the feasibility of ontology-driven, grid-integrated energy management in real-world energy communities.

Figure 13 illustrates the 24-hour closed-loop operation of the BESS, divided into three subplots for clarity.

1. Shows the forecasting process. The GCM module generates **forecasted limits (1)** for the battery as part of the HESS, based on one-week-ahead fore-casts. These limits are then provided to the EMS4HESS module, which computes the planned flexibility deployment for the next 24 hours.

2. Displays the **planned flexibility deployment (2)** and the **current limits (3)**, calculated by the Clipper module. This ensures that EMS4HESS **setpoints (4)** remain within a safe margin, computed online using the latest grid measurements. If violations occur, EMS4HESS setpoints are curtailed, resulting in adjusted Clipper setpoints. Note that these safe margins differ from those computed by GCM.

3. Depicts the complete 24-hour closed-loop operation, including **forecasted limits (1)**, **planned flexibility (2)**, **current limits (3)**, **calculated setpoints (4)**, and the **measured active power from the BESS**. The plot demonstrates that the battery closely follows the Clipper setpoints, with only minor deviations.

In addition to a range of technical and economic Key Performance Indicators (KPIs), self-sufficiency and self-consumption have been identified as two of the most relevant indicators for the PARMENIDES energy communities. Initial evaluations using GCM and EMS4HESS have demonstrated the significant potential of optimal utilisation of the HESS. Table 5 presents results for a representative sunny day in November, comparing scenarios without and with consideration of the available BESS.

Table 5: Analysis of selected PARMENIDES KPIs.

KPI	Without BESS	With BESS
Total Export [kWh]	142.5	2.5
Total Import [kWh]	144.0	29.0
Self-sufficiency [%]	59	94
Self-consumption [%]	60	99

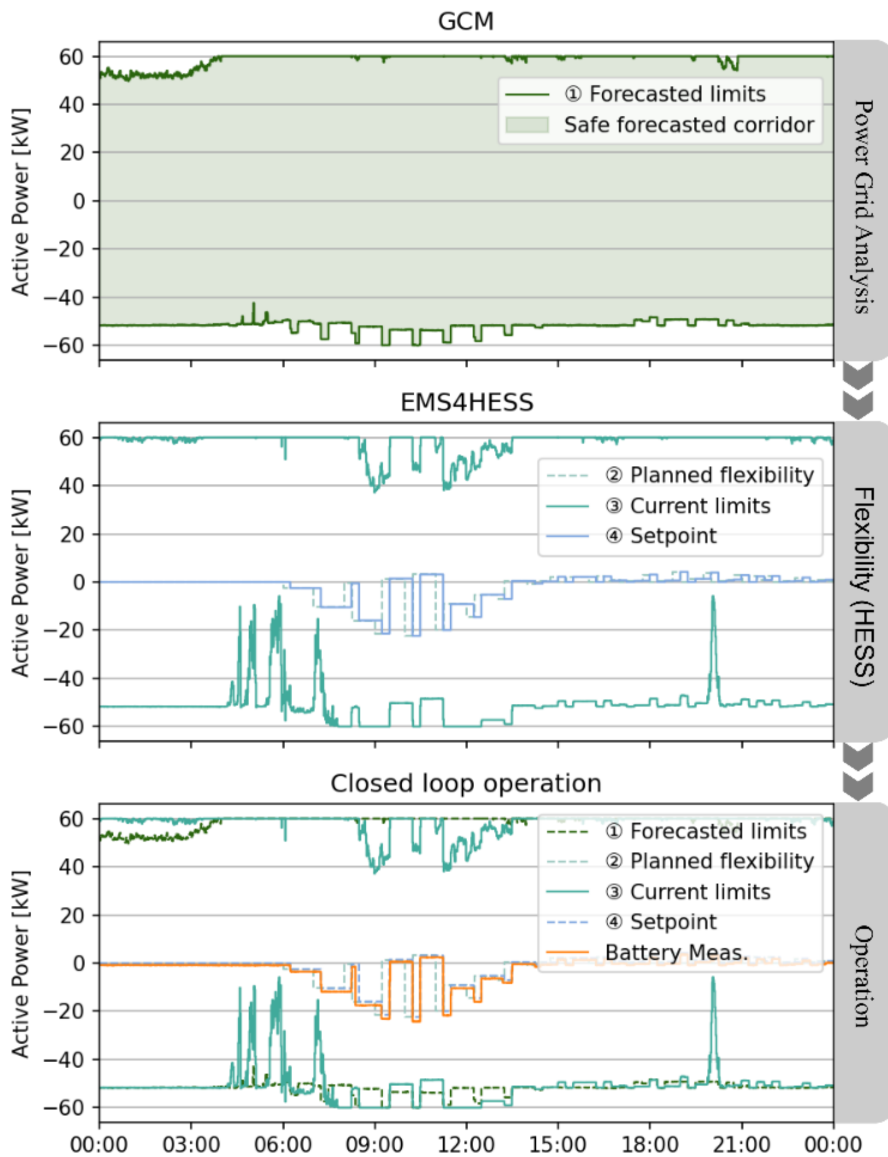


Figure 13: Closed-loop operation of the battery, showing forecasted limits (1), planned flexibility (2), current limits (3), calculated setpoints (4), and measured charging power of the BESS.

The significant reduction in grid imports and exports observed in the pilot – dropping exports from 142.5 kWh to 2.5 kWh and imports from 144 kWh to 29 kWh – has direct implications for both grid stability and economic performance.

Lower exports reduce reverse power flows, mitigating voltage rise and thermal stress on low-voltage feeders, which in turn decreases the need for costly grid reinforcements. Similarly, reduced imports alleviate peak demand, supporting distribution system operators in maintaining operational limits without additional infrastructure investment. From a cost perspective, higher self-sufficiency (94 %) and self-consumption (99 %) enable community members to maximise the use of locally generated renewable energy, lowering exposure to retail electricity prices and grid tariffs. This optimisation translates into tangible savings for prosumers while fostering a more resilient and decentralised energy system.

The observed improvements in self-sufficiency and self-consumption highlight the potential of ontology-driven energy management systems to significantly enhance the operational efficiency of RECs. By reducing grid imports and exports, the EMS4HESS approach not only optimises local energy use but also alleviates stress on distribution networks, contributing to grid stability and cost savings for community members. However, it is important to note that these results are based on a single representative day under favourable conditions. While this short-term evaluation demonstrates the feasibility and benefits of the proposed architecture, a comprehensive long-term assessment is planned in future projects. This extended analysis will capture seasonal variations, diverse load profiles, and dynamic market conditions to validate scalability and robustness across different operational contexts.

4 Results from the Swedish Pilot

The results of the Swedish pilot are a combination of physical and digital infrastructure, which together can be used to examine different energy storage technologies, their control and operation as a HESS, and evaluating their flexibility potential. Figure 14 below illustrates a summary of these outcomes.

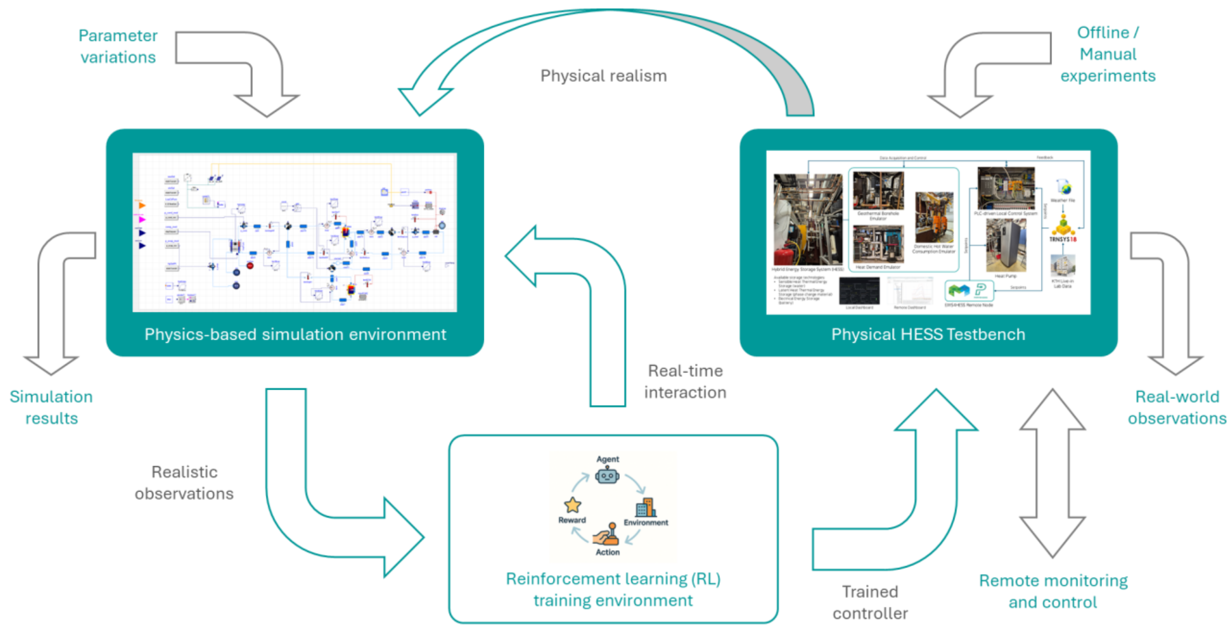


Figure 14. Outcomes of the Swedish pilot.

The digital infrastructure is largely responsible for developing and demonstrating the PARMENIDES Flexibility Strategy, while the physical HESS Testbench provided physical realism to the simulation environment used in training a control agent. Both the physical and digital infrastructure could also be used independently. For instance, the HESS Testbench was used to emulate demand from the KTH Live-in Lab and geothermal borehole temperatures. The simulation environment was also developed in successive iterations using different configurations.

4.1 Demonstration of PARMENIDES Flexibility Strategy

The demonstration of the PARMENIDES Flexibility Strategy focuses on the “Flexibility Strategy III: Exploration/exploitation trade-off” defined in the PARMENIDES project proposal. To recap, along with the aim to maximize flexibility, end-users are allowed to choose among different modes: “economy”, “comfort”, and “environment”. The PARMENIDES Flexibility Strategy builds on the premise that pursuing an objective always implies trade-offs on the others.

It is worth mentioning that every iteration of the simulation to demonstrate the PARMENIDES Flexibility Strategy (see Table 1) yielded results that were useful in the succeeding iterations. The key results that led to changes in the next models are summarized in Table 6 below.

Table 6. Key results in simulation-based iterations and corresponding changes

Model Iteration	Key Results and Limitations	Changes vs Previous Iteration
Model 0	Proposed and demonstrated the application of "Curriculum-based Reinforcement Learning" as a viable and appropriate approach to achieve the objective of the PARMENIDES Flexibility Strategy. Results presented at 2026 IEEE PES-IM and published through a conference paper[8] Limitations: • Co-simulation with EnergyPlus is callback-based, meaning EnergyPlus controls time advancement. As a result, the RL loop (training script) must synchronize to callback triggers rather than stepping the simulation directly, limiting access to intermediate states and explicit timestep control. • EnergyPlus is less flexible if only lumped parameters such as U-values, heat capacitances, and high-level building details are available. It relies on detailed building specifications, which, if unavailable could lead to inaccurate simulations. • The use of EnergyPlus required the software to be present in the machine, making it less flexible • Used an electric heater for space heating • Only comfort and economy mode were tested	
Model 1	Affirmed the approach from Model 0 and demonstrated results of flexibility activation aligned with USEF flexibility trading protocol message structure. A more detailed trade-off analysis was performed. Established the full machine learning training pipeline based on a custom environment with functional mock-up unit (FMU) co-simulation. All preference modes were tested (comfort, economy, environment). Basis for the simulation and RL training in models 2 and 3.	Transitioned building the system model in Dymola (Modelica) and exporting it to an FMU. Changed the system design to approximate the HiL set-up in KTH Granryd Laboratory.

[8] L.R. Payonga, H. Madani, M. Stefan, "Curriculum-based Reinforcement Learning for Flexibility Evaluation in Buildings," in *Proceedings of the 2026 IEEE Power and Energy Society International Meeting*

	Results to be presented at 15th IEA Heat Pump Conference and will be published as a conference paper[9]	
Model 2	Control signals were changed to command sets (e.g., charge SHTES, discharge SHTES; see Table 3) rather than independent signals such as in Model 1 (i.e., heat pump setpoint and load circulation pump modulation). This is a result of earlier simulations/training runs where exploratory control of the heat pump, three-way valves, and circulation pump led to unsafe states and combinations. Considering that results from the HESS testbench became available around the same time, RL training for Model 2 was skipped in favor of improving and completing the system model in Model 3.	
Model 3	Control signals are comprised of the TES command set ("TES state") as in Model 2, along with battery activation, and lower and upper limits to the battery SOC. In this model, the command sets are implemented by a deterministic logic and sequence of operations and interlocks within the Modelica-based FMU based on HESS testbench experiment results. Interval for new commands was changed to 30 minutes to account for realistic heat pump operation.	Implemented revisions in the Modelica system model based on the results of storage technologies characterization from the HESS Testbench.

4.1.1 Application of reinforcement learning (RL) method

Reinforcement Learning (RL) is a machine learning method that involves training an *agent* to learn a *policy* that pursues a predefined objective by interacting with an *environment* through its *actions* based on *state* observations. Desirable outcomes based on actions are incentivized with *rewards*, further reinforcing the action trajectories. It is called Deep RL when the approach is performed to tune a deep neural network (DNN) used to represent a policy in high-dimensional problems and control tasks. This approach is ideal for systems whose dynamics are too complex to model explicitly and/or are exposed to uncertainties and volatility. Developing the *training environment*, specifying the *state space*, *action space*, and *reward functions* as well as defining *evaluation* and *model selection criteria* are core activities in RL. Several RL algorithms have been proposed, with Proximal Policy Optimization (PPO) being one of those known for reliability and stability.

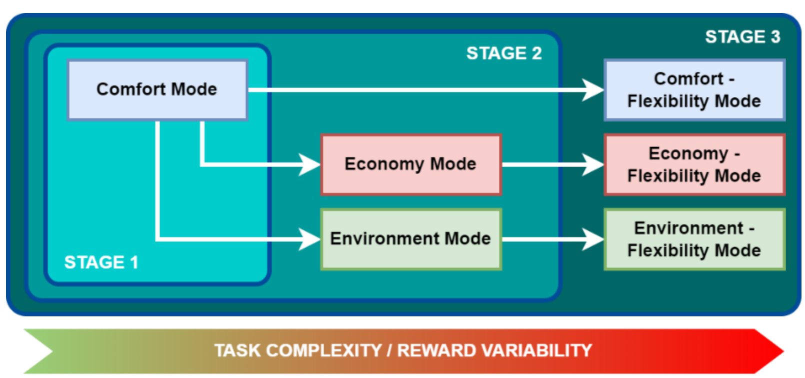
This approach was chosen for implementing the PARMENIDES Flexibility Strategy because RL explicitly addresses the exploration-exploitation trade-off inherent to sequential decision-making problems. During

[9] L.R. Payonga, H. Madani, "Evaluating heat pump-enabled building flexibility potential via deep reinforcement learning and trade-off analysis," in *Proceedings of the 15th IEA Heat Pump Conference*

training, the agent explores different control actions while progressively biasing its policy toward actions that yield higher expected cumulative reward. This behavior critically depends on the specification of the reward function. For instance, the reward can be designed to positively reinforce flexibility actions such as load shedding, potentially achieved through discharging thermal energy storage, while penalizing excessive deviations from predefined comfort temperature bounds. Through RL, the control agent can thus learn action trajectories that balance these competing objectives, maximizing flexibility provision without allowing comfort penalties to outweigh the associated benefits.

One challenge encountered during training was that flexibility requests arising from grid congestion can be considered rare events over typical operational horizons. This challenge is compounded by the volatility of electricity prices, which directly affects the cost-minimization objective, as well as by the requirement to maintain indoor comfort within acceptable bounds regardless of other optimization preferences. From an RL perspective, these factors pose challenges to stable and efficient policy convergence. Consequently, in addition to *reward shaping* through careful design of the reward function and the selection of an appropriate RL algorithm (in this case, PPO), *state augmentation* and *curriculum learning* techniques were employed[10,11]. State augmentation was used to enrich the state representation with additional operational variables that better capture system dynamics and restore the Markov property, while curriculum learning was applied by progressively exposing the control agent to increasingly challenging scenarios rather than requiring it to address all objectives simultaneously from the outset.

The curriculum learning approach starts with training the agent to maintain comfort, then the saved model gets trained again to pursue cost or carbon emissions reduction, and then finally exposed to flexibility requests in the final phase. In the final version of the HESS model, the reward functions included incremental rewards in keeping the SOC of the battery at least 80% and the temperature of the tanks at least 30°C.



[10] L.R. Payonga, H. Madani, M. Stefan, "Curriculum-based Reinforcement Learning for Flexibility Evaluation in Buildings," in *Proceedings of the 2026 IEEE Power and Energy Society International Meeting*

[11] L.R. Payonga, H. Madani, "Evaluating heat pump-enabled building flexibility potential via deep reinforcement learning and trade-off analysis," in *Proceedings of the 15th IEA Heat Pump Conference*

Figure 15. Three-stage curriculum learning approach.

Simulations in Modelica and storage tests conducted in the HESS testbench motivated a revision of the original direct component control strategy implemented in Model 0 and Model 1 (see Table 6) toward a hierarchical control architecture. In this approach, the RL agent is restricted to learning high-level control actions (e.g., charging or discharging commands), while low-level component actuation is handled by deterministic, rule-based control logic. This design choice was determined to be the safest both from a training perspective – where RL inherently involves exploration of random and potentially unsafe actions and state transitions – and from a physical implementation perspective, where misaligned or conflicting low-level controls (e.g., valve interlocks) could lead to undesirable operating conditions such as pressure build-up or flow reversals. The suggested logic and sequence of operation are detailed in section 4.2.3.

The complete RL training and co-simulation framework is illustrated in Figure 16[12]. A custom *Gymnasium*-based[13] training environment, `BuildingEnv`, was developed to interface with an FMU exported from the Modelica system model via the *FMPy* API[14]. Within `BuildingEnv`, the observation and action spaces, as well as the reward functions, are defined. In addition, auxiliary methods are implemented to manage FMU interactions, process simulation outputs, integrate external data sources, and perform state augmentation.

The overall training and evaluation workflow is orchestrated by a main script in which instances of `BuildingEnv` are created. This script also employs the PPO algorithm as implemented in the Stable Baselines3 framework[15] to train the control policy. Upon completion of training, the learned model, represented by the neural network's weights and biases, is saved and can in principle be deployed either within the FMU environment or on a physical system. Within the scope of PARMENIDES, deployment was limited to the FMU-based simulation environment. State and action spaces are detailed in Table 7.

---

[12] Partly based on Bousnina D, Guérassimoff G. Optimal energy management in smart energy systems: A deep reinforcement learning approach and a digital twin case-study. Smart Energy 2024;16:100163

[13] https://gymnasium.farama.org/

[14] https://github.com/CATIA-Systems/FMPy

[15] https://stable-baselines3.readthedocs.io/en/master/

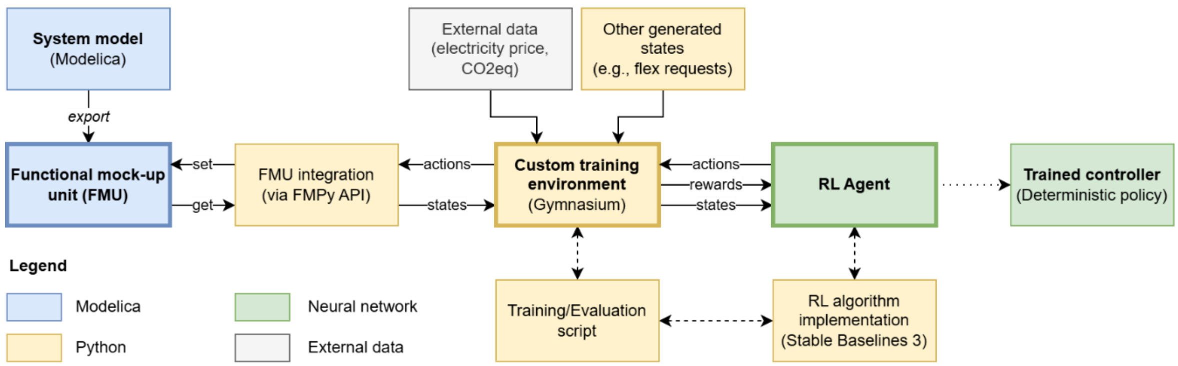


Figure 16. RL training and co-simulation framework.

Table 7. State and action spaces in the model iterations

<table>
<tr><th>Model</th><th colspan="3">State observations</th><th>Actions (options)</th></tr>
<tr><td>0-3</td><td colspan="3" rowspan="2">Thermal: indoor temperature
Electrical: electricity demand, change in electricity demand
Price: electricity price, change in electricity price
Weather: outdoor temperature
Flexibility: flexibility request signal, forecast deviation
Settings: mode, indoor temperature setpoint, nominal electricity demand, reference electricity price</td><td></td></tr>
<tr><td>0</td><td>Heater operation (2: on/off)</td></tr>
<tr><td>1</td><td rowspan="3">Temporal: time of day and day of year (cyclically encoded)
Thermal: change in indoor temperature, supplied thermal energy to the load, tank(s) temperature
Flexibility: flexibility limits (min, max)
Device status:
Load circulation pump status
Settings: tank(s) temperature setpoint</td><td></td><td rowspan="2"></td><td>Heat pump heat curve offset (7: -3 to +3),
Load circulation pump modulation (11: 0 - 1)</td></tr>
<tr><td>2</td><td rowspan="2">Thermal: thermal energy supplied by the heat pump to each of the tanks, thermal energy supplied by tanks to load
Device status: storage selector valve status, source selector valve status</td><td>TES command (4: SHTES/LHTES charge/discharge)</td></tr>
<tr><td>3</td><td>Electrical: PV output, apartment electricity demand (non-heating), battery charge/discharge energy, battery SOC</td><td>TES command (5: SHTES/LHTES charge/discharge, “do nothing”),
Battery operation (2: activate/deactivate grid connection)
Battery lower SOC
(3: 0.2/0.35/0.5)
Battery upper SOC
(3: 0.7/0.8/0.9)</td></tr>
</table>

In-training evaluation was done according to relevant KPIs: electricity cost and effective rate, equivalent CO_2 emissions, temperature setpoint deviation and spread, flexibility response rate, and flexibility provided. A model selection algorithm is in place, which selects a model (i.e., neural network parameters) depending on the optimization mode (comfort/economy/environment) and the stage in the curriculum.

4.1.2 Flexibility request generation and response

Upon reaching Stage 3 of the control agent curriculum, the agent is exposed to flexibility requests and is rewarded according to the proportion of flexibility requests responded to, as well as the total amount (in terms of energy) of flexibility provided.

For the purposes of PARMENIDES, the implementation of flexibility evaluation and activation is illustrated in Figure 17. The flexible models from Stage 3 are deployed, depending on the user-centric optimization mode (i.e., comfort, economy, or environment). The selected model (i.e., trained controller) is used to manage the operation in the digital replica in deterministic mode (i.e., no exploration); the output of which is the status quo forecast, which corresponds to the "D-prognosis" in the USEF Flexibility Trading Protocol (UFTP). The forecast, together with a congestion schedule and the corresponding flexibility allocation represented by minimum and maximum bounds of demand for the customer/energy community, are input to the *FlexRequest processor*, which computes the allowable deviation limits from the status quo forecast. These deviation limits then form part of the observation space of the flexible model in operation, which will maximize flexibility activation without compromising the main optimization preference.

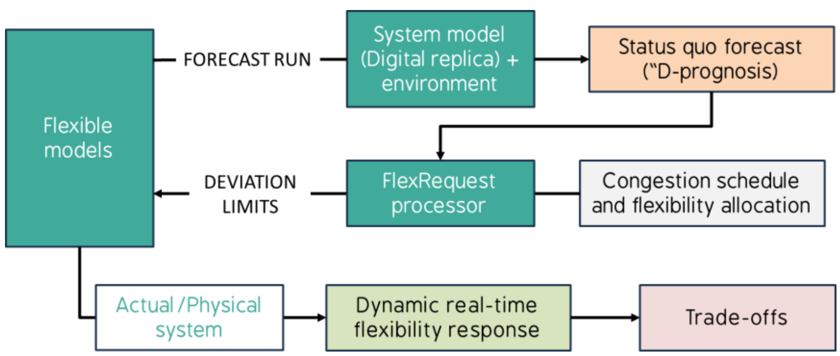


Figure 17. Flexibility request generation and controller response

In Figure 17, the flexible model is shown to be deployed in an actual/physical system to make it decide real-time; but the same can also be done to predict how much flexibility will be offered by the energy community to the Flexibility Requesting Party (FRP).

4.1.3 Implementation of Use Cases 3 and 4: Automated (with Human Inputs)

The approach implemented digitally in the Swedish Pilot could fall under Use Cases 3 and 4, in that the optimization preference is a "human input" (UC 3), which if programmed to execute automatically in an EMS, falls under UC 4. If an intermediate process of evaluating the predicted flexibility is required prior to activation, that would fall under UC 3.

Table 8 below shows the results of Model 1 (SHTES tank + building thermal mass) deployment in the simulation environment. In this specific demonstration, the simulated building was subjected to an extreme test case of a fixed daily 7-hour congestion schedule for 3 months (January to March) from 15:00 to 22:00 and was allocated a range of 0 to 500 Wh. Note that for this simulation, only the heat pump electricity use was accounted for. Figure 18 shows an instance when the trained controller was able to respond to most of the test flexibility requests.

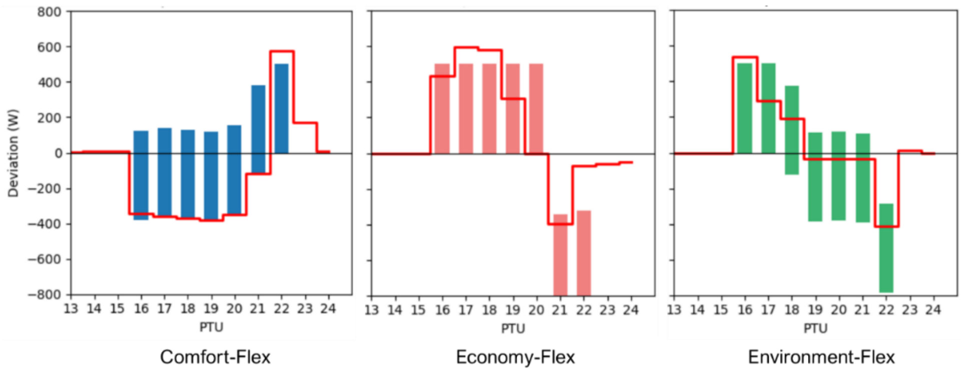


Figure 18. Demonstration of response to flexibility requests.

The results showed that while preference-optimized models (comfort, economy, environment) generally achieved their objectives, trade-offs were apparent when responding to flexibility requests (comfort-flex, economy-flex, environment-flex). For instance, electricity costs varied from 0 to 8%, equivalent CO2 emissions by 2 to 5%, while comfort remained within a reasonable range (i.e., still < 1K).

Table 8. KPIs based on operation modes

KPI	Comfort	Comfort-Flex	% change	Economy	Economy-Flex	% change	Environ-ment	Environ-ment-Flex	% change
Total electricity cost (SEK)	620.30	622.51	0	458.45	488.50	7	511.15	552.16	8
Effective electricity rate (SEK/kWh)	0.65	0.63	-2	0.57	0.58	1	0.61	0.63	3
Equivalent CO2 emissions (kgCO2e)	26.26	26.79	2	21.69	23.00	6	22.65	23.68	5
Effective CO2e emissions (gCO2e/kWh)	27.31	27.29	0	27.07	27.09	0	27.07	27.12	0
Average setpoint deviation (K)	0.23	0.50	117	0.53	0.00	-100	0.49	0.58	18
90% central range (K)	1.90	2.82	48	6.91	6.67	-3	3.18	2.60	-18
Total electricity use (kWh)	961.67	981.59	2	801.00	849.19	6	836.69	872.99	4
Peak demand (W)	912.30	912.30	0	1049.19	1080.78	3	954.98	943.47	-1
Flexibility activation rate (%)		63.17			15.40			63.33	

To have a broader view of the trade-offs in KPIs for each optimization mode, regression plots are shown in Figure 19.

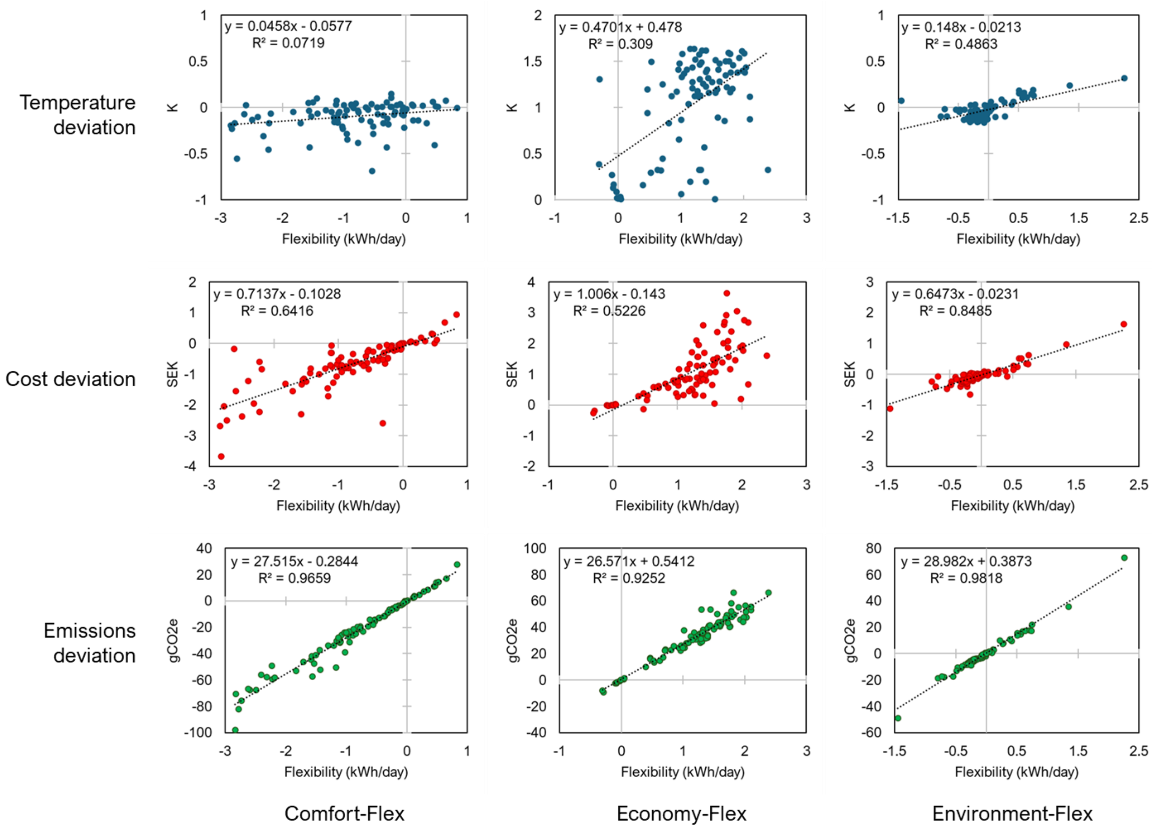


Figure 19. Regression plots of changes in KPIs relative to daily flexibility provision.

Although the full HESS testbench digital replica (Model 3) was finished within the project duration, it was not possible to test and implement the PARMENIDES Flexibility Strategy in it. It is, however, expected that flexibility activation as well as their trade-offs will become more evident in such an implementation.

4.2 HESS hardware-in-the-loop testbench

4.2.1 Simulation and emulation of KTH Live-in Lab profiles

Heating demand from the KTH Live-in Lab was used as input to a Trnsys model that outputs setpoints that the PLC-driven system would attain to reach as part of the emulation process. The control panel for the HESS Testbench (shown in automatic mode for load profile and ground source emulation) and the Trnsys model are in Figure 20.

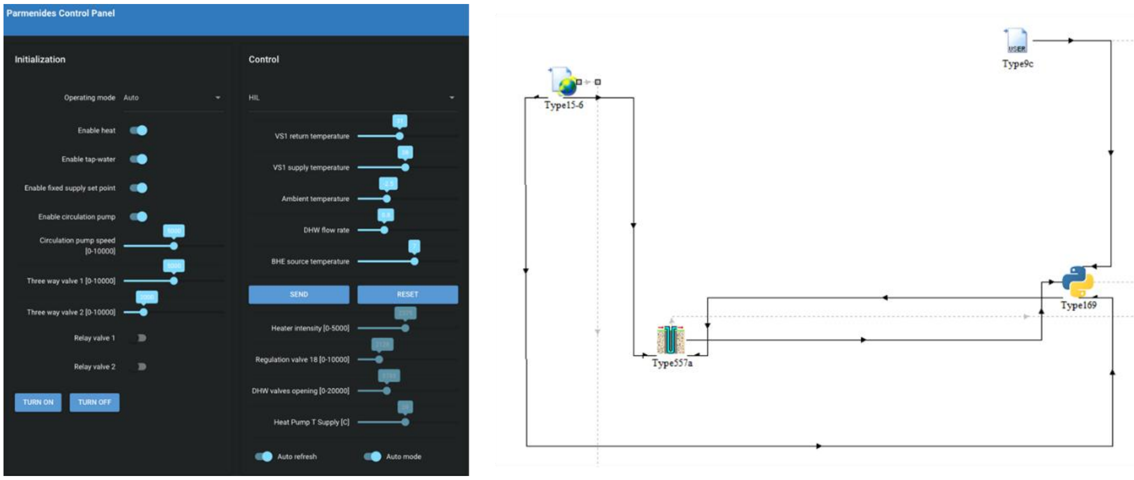
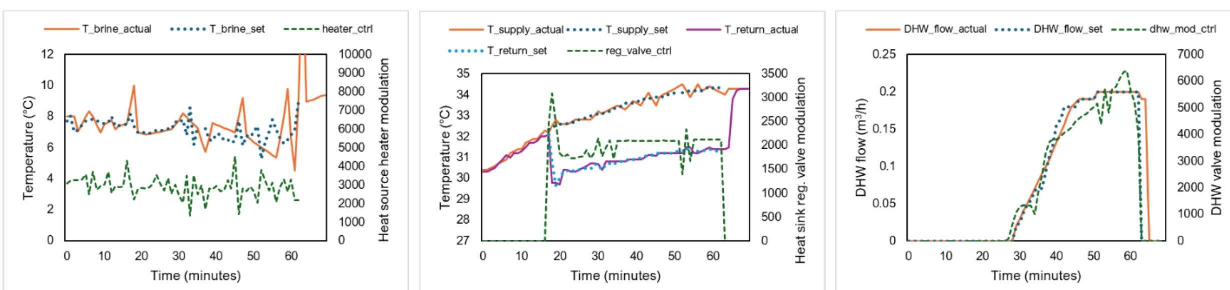


Figure 20. Simulation and emulation control of the HESS Testbench

Figure 21 shows how the HESS Testbench emulates the results of the simulation. For instance, in this demonstration, the heat source emulator heater in Figure 21-A is modulated automatically to have T_brine_actual closely track T_brine_set.[16] In a similar manner, in Figure 21-B, heat rejection to the district cooling network is modulated with a regulating valve to emulate heating demand, such that target return temperatures are attained according to the Trnsys simulation. Lastly, in Figure 21-C a motorized valve[17] is also automatically modulated to track the DHW flow determined by the simulation.



A) Heat source emulation

B) Heat demand emulation

B) DHW use emulation

Figure 21. Demonstration of emulation by the HESS Testbench

4.2.2 Integration with EMS4HESS

The internal Energy Management System (EMS) hierarchy was mapped by systematically translating the energy network schemas developed in the Granryd Lab Swedish pilot into a hierarchy tree that connects the assets which are in the scope of the PARMENIDES project use cases. Starting from the physical layer, assets such as PVs, energy storage devices, loads, and metering points were mapped into functional

[16] Plot of heater_ctrl shrunk for legibility. Actual range is only 0-5000.

[17] Two motorized valves are available to support emulation of larger flows

groups, and were organized into progressively higher abstraction levels reflecting the site layers. This bottom-up decomposition allowed the preservation of the electrical and communication relationships defined in the original network while introducing clear boundaries for monitoring and optimization functions.

The actual EMS hierarchy can be visualized in Figure 22 below, through a dedicated Grafana panel. Each level – from individual devices and field components up to pilot, subsite and site – is represented to reflect both operational dependencies and data flows. This layout allows operators to quickly navigate between granular measurements and higher-level measurements within the same interface, since any of the listed resources can be related to a set of properties.

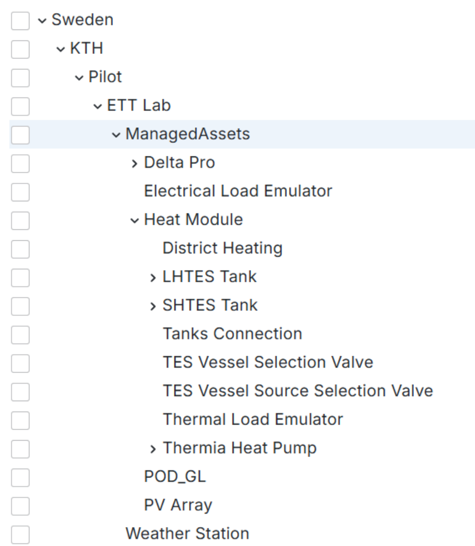


Figure 22. EMS hierarchy for Swedish Pilot

To illustrate how live feed data flows are operationalized within the EMS, the following Excel snippets in Figure 23 present an example subset of properties and their corresponding mappings to the remote node via the MQTT connector. The first snippet lists the modelled EMS properties, including naming conventions, units, and logical grouping within the hierarchy.

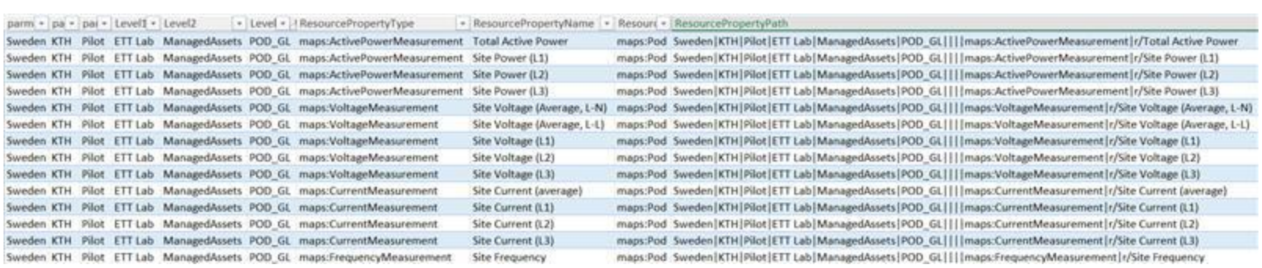

<table>
<tr><th>parm</th><th>pa</th><th>pai</th><th>Level1</th><th>Level2</th><th>Level</th><th>ResourcePropertyType</th><th>ResourcePropertyName</th><th>Resour</th><th>ResourcePropertyPath</th></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:ActivePowerMeasurement</td><td>Total Active Power</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:ActivePowerMeasurement|r/Total Active Power</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:ActivePowerMeasurement</td><td>Site Power (L1)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:ActivePowerMeasurement|r/Site Power (L1)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:ActivePowerMeasurement</td><td>Site Power (L2)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:ActivePowerMeasurement|r/Site Power (L2)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:ActivePowerMeasurement</td><td>Site Power (L3)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:ActivePowerMeasurement|r/Site Power (L3)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:VoltageMeasurement</td><td>Site Voltage (Average, L-N)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (Average, L-N)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:VoltageMeasurement</td><td>Site Voltage (Average, L-L)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (Average, L-L)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:VoltageMeasurement</td><td>Site Voltage (L1)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L1)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:VoltageMeasurement</td><td>Site Voltage (L2)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L2)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:VoltageMeasurement</td><td>Site Voltage (L3)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L3)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:CurrentMeasurement</td><td>Site Current (average)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (average)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:CurrentMeasurement</td><td>Site Current (L1)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L1)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:CurrentMeasurement</td><td>Site Current (L2)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L2)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:CurrentMeasurement</td><td>Site Current (L3)</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L3)</td></tr>
<tr><td>Sweden</td><td>KTH</td><td>Pilot</td><td>ETT Lab</td><td>ManagedAssets</td><td>POD_GL</td><td>maps:FrequencyMeasurement</td><td>Site Frequency</td><td>maps:Pod</td><td>Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:FrequencyMeasurement|r/Site Frequency</td></tr>
</table>

Figure 23. Properties and mappings to the remote node

While the second shows the communication-layer configuration that binds each EMS property to its MQTT topic. Together, these mappings clarify how field measurements are standardized inside the EMS.

ResourcePropertyPath	topic
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (Average, L-N)	ParmenidesEdgeSweden/Local/EM/r/voltageAvgLN/V
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (Average, L-L)	ParmenidesEdgeSweden/Local/EM/r/voltageAvgLL/V
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L1)	ParmenidesEdgeSweden/Local/EM/r/voltageL1/V
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L2)	ParmenidesEdgeSweden/Local/EM/r/voltageL2/V
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:VoltageMeasurement|r/Site Voltage (L3)	ParmenidesEdgeSweden/Local/EM/r/voltageL3/V
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (average)	ParmenidesEdgeSweden/Local/EM/r/currentAvg/A
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L1)	ParmenidesEdgeSweden/Local/EM/r/currentL1/A
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L2)	ParmenidesEdgeSweden/Local/EM/r/currentL2/A
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:CurrentMeasurement|r/Site Current (L3)	ParmenidesEdgeSweden/Local/EM/r/currentL3/A
Sweden|KTH|Pilot|ETT Lab|ManagedAssets|POD_GL|||maps:FrequencyMeasurement|r/Site Frequency	ParmenidesEdgeSweden/Local/EM/r/frequency/Hz

Figure 24. Binding of EMS properties to MQTT topics

On the Swedish pilot side, the architecture is organized around a remote node that acts as the integration and point between the ETT Lab energy assets and the central EMS4HESS. This node hosts a set of protocol-specific connectors – such as MQTT for message-based exchange and REST API interfaces for specific systems like Enphase (for the PV), Ecoflow (for the battery), and Accio (for the weather station) – each responsible for normalizing device data and rerouting it to the EMS4HESS. By decoupling field communication from the EMS core, the remote node enables modular onboarding of heterogeneous communication standards. This connector-based design ensures interoperability and scalability while maintaining a clear boundary between site-level device control and centralized aggregation and elaboration functions.

4.2.3 Energy storage tests

The three energy storage technologies – SHTES, LHTES, and battery – were tested at the HESS Testbench to characterize their properties, estimate flexibility potential, and determine operational considerations. Highlights of the test are shown in Figure 25.

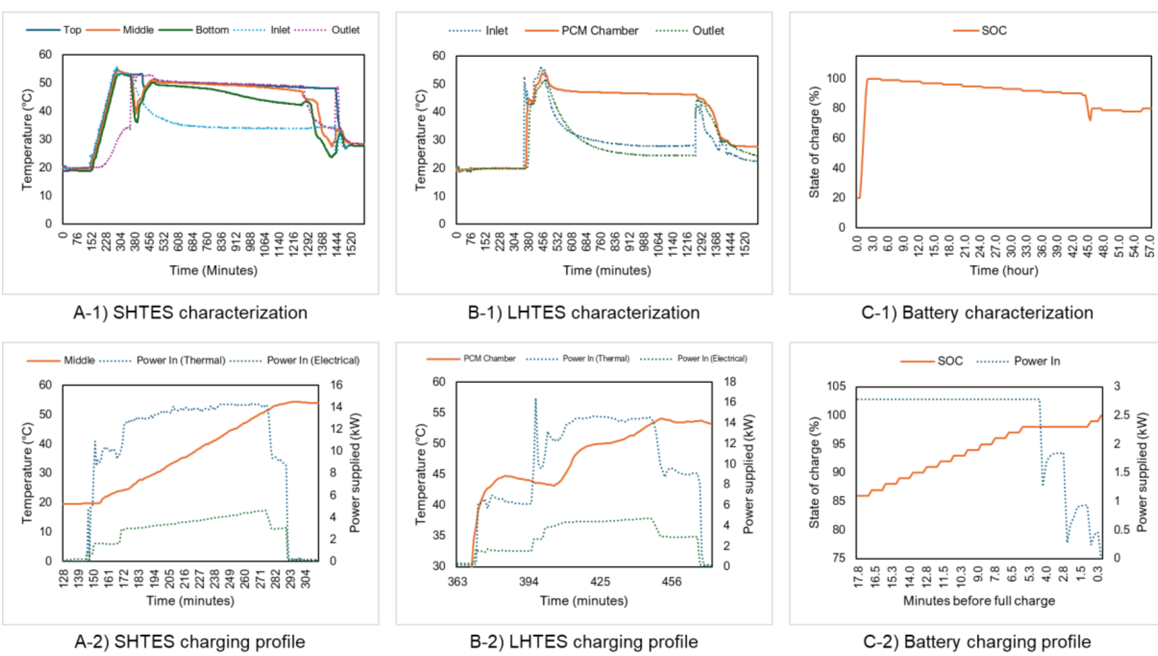


Figure 25. Energy storage tests at the HESS Testbench

The first row in Figure 25 shows a complete round of tests from charging to discharging. The second row highlights certain properties relevant for fine-tuning the specifications of the simulation environment. Figures 17-A and 17-B shows both the electrical and power supplied to each TES. As is expected for a heat pump, the thermal energy injected into the TES vessels is more than the electric energy drawn from the grid. The stepped ramping up of the heat pump compressor is also evident in the power plots.

In Figure 25-A1, the SHTES's temperatures throughout its top, middle, and bottom layers show stratification. It was possible to observe these properties because of the thermocouples attached to the SHTES tank. A closer look at its charging profile in Figure 25-A2 shows the expected linear charging characteristics for an SHTES. Meanwhile, it was only possible to observe the middle portion of the LHTES tank (see Figure 25-B1) because there was only one thermocouple attached to it. Nonetheless, the set-up was such that water flowed only from top to bottom (see Figure 4) so the inlet and outlet temperatures could be representative of the top and bottom temperatures when there is flow. Stratification within the tank and internal temperatures of the PCM could not be captured, however. Zooming into the charging characteristics of the LHTES tank in Figure 25-B2 shows a latent heat region, although it did not fall within the expected band of 45 to 48°C melting temperature as specified. This could be due to the backflow from the SHTES tank, which is apparent around the time the LHTES started charging. A similar characteristic was observed during discharge. This observation led to the addition in the simulation environment of an intermediate 3-way valve at the junction among the SHTES, LHTES, and the load loop (see Figure 2), which has an interlocked logic depending on the operation mode.

As regards the battery, the only measurement that could be obtained successfully from the Enphase API was the SOC. However, a live data monitoring of power in and out could be done through the Enphase mobile app. In Figure 25-C1, four (4) operation segments could be observed. The charging phase showed a linear ascent of the SOC. The battery showed self-discharge of 0.25% of its SOC per hour. In the setting demonstrated, the upper SOC was changed to 80% from 100%. Thus, when the battery was discharged with a fixed load, it started charging when SOC was below 80%. The battery system maintains 80% SOC by constantly re-charging when SOC is at 78% even without load. In Figure 25-C2, when the upper SOC limit is 100%, it was observed that in the last phase of charging the power draw is ramped down in step increments. As a safety measure, the battery also cuts off supply to the load when the battery has been discharged to the lower SOC threshold.

The abovementioned tests led to the refinement of the Modelica-based HESS simulation environment. These include mimicking the stepped operation of the heat pump compressor with around 20-minute hold in an intermediate compressor speed, the interlocks of heat pump circulation pumps, the ramp times of the valves, pump specifications according to performance curves, battery charging and discharging behavior, and what were deemed safe operating configurations and sequences.

Specifically for the thermal subsystem, apart from the operating modes outlined in Table 3 and illustrated in Figure 3, it was important to perform a sequence that allows proper establishment of flow in relevant segments and avoiding pressure build-up especially during valve actuations.

4.2.4 Implementation of Use Case 2: Manual activation (Active)

Based on the results of the energy storage tests, the following table shows the characteristics of the HESS Testbench.

Table 9. Energy storage characteristics of the HESS Testbench

Property	SHTES	LHTES	Battery	**HESS**	Note
Charging range	20-50°C (30 K ΔT)		20-100% SOC		
Charging time (mins)	133	91	90	**224**	Battery charging can be done simultaneously with the thermal subsystem
Peak electrical charging power (kW)	4.67	4.7	2.78	**7.48**	Vessels are charged one at a time and capped by the heat pump peak power; Nameplate power draw by battery is 3 kW, so value can at least be 7.70.
Electrical energy in (kWh)	7.39	4.45	4.16	**15.99**	
Peak thermal charging power (kW)	14.4	16.45	N/A	**16.45**	
Thermal energy in (kWh)	28.48	14.80	N/A	**43.27**	
Electricity flexibility potential (kWh/h)	3.33	2.93	2.77	**4.28**	
Thermal flexibility potential (kWh/h)	12.85	9.75	N/A	**11.59**	
Energy capacity (kWh)	43.27		3.6	**46.87**	

Under Use Case 2, users/energy community participants can be informed of their flexibility potential or a recommended action, but there will be no direct control of assets. In this case, this particular exercise of identifying the flexibility potential of the HESS Testbench fits within such a framework. Nonetheless, the same information can be used even in automated activations in Use Cases 3 and 4, although they will rather be more dynamic and driven by the PARMENIDES Flexibility Strategy.

5 Replicability and Scalability Analysis

PARMENIDES focuses on the development, deployment, and validation of innovative, interoperable, and secure concepts for providing flexible energy services based on Hybrid Energy Storage Systems (HESS). These systems act as a virtual abstraction layer integrating heterogeneous energy storage and flexibility resources, including battery storage systems, hydrogen storage, heat pumps, buildings, district heating networks, and electric vehicles. The overall objective is to enable coordinated, scalable, and secure flexibility provision across multiple energy domains and stakeholder contexts.

The Scalability and Replicability Analysis (SRA) in PARMENIDES aims to demonstrate that the developed solutions are not only functional at the piloting sites but are scalable to larger communities and replicable across different technical and regulatory contexts. It is designed to evaluate the technical, economic, and social potential of these solutions by emphasizing scalability, replicability, and performance evaluation across heterogeneous technical setups and regulatory environments. Such a setup and analysis are essential to ensure that PARMENIDES results go beyond isolated demonstrations and can be transferred to other contexts.

The AIT Virtual Laboratory (VLab) and its SRA framework play a central role in supporting this objective by enabling systematic, controlled, and repeatable experimentation beyond physical pilot constraints.

5.1.1 AIT VLab Concept and Virtual Energy Community Testbed

The AIT VLab provides a virtualized experimentation environment based on containerized mock-ups of energy assets, including smart meters, photovoltaic systems, battery storage, electric vehicle charging infrastructure, and measurement devices. These virtual assets form the foundation of a virtual energy community testbed that complements physical pilot sites. The testbed allows PARMENIDES solutions to be evaluated under controlled yet realistic conditions, enabling analysis of technical scalability, controller behaviour, and system-level performance without the limitations of physical deployment scale.

The virtual energy community is designed to support controller-in-the-loop testing, where real control algorithms interact with emulated assets and communication infrastructures. This approach enables early identification of performance bottlenecks, communication constraints, and control stability issues while preserving a high degree of realism. In addition, the VLab enables the systematic evaluation of energy-related, economic, and ICT-related Key Performance Indicators (KPIs), ensuring a holistic assessment of solution performance.

5.1.2 Scalability Assessment Methodology

The AIT VLab SRA is based on a structured methodology that incrementally increases both the size and complexity of the emulated energy community. Scalability is assessed by varying the number of prosumers in the virtual environment, ranging from small-scale setups (e.g., 10 prosumers) to large-scale communities with hundreds or thousands of participants. In parallel, system complexity is increased by adjusting messaging frequencies, communication patterns, and ICT characteristics, such as network latency or bandwidth constraints.

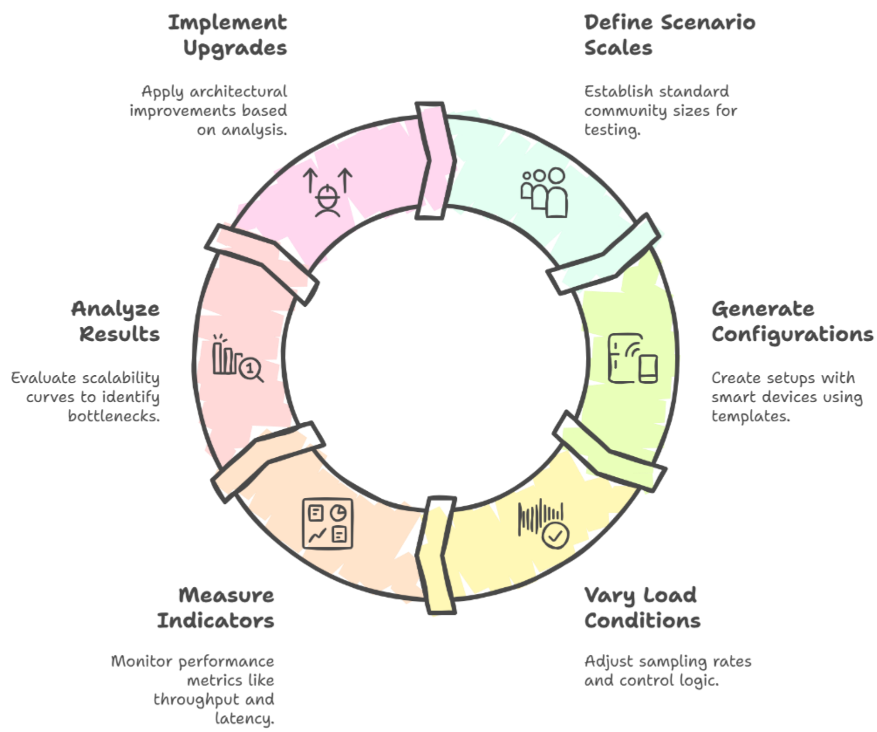


Figure 26: Overview of the AIT VLab-based scalability assessment methodology.

Throughout these experiments, a comprehensive set of KPIs is monitored to evaluate system performance and limits. These KPIs include message throughput, latency, memory consumption, CPU usage, and the maximum number of supported prosumers. By systematically correlating system size and complexity with observed KPIs, the VLab SRA provides quantitative evidence of how PARMENIDES solutions scale under increasing operational demands. This directly supports the SRA objectives by demonstrating that the developed concepts remain functional and efficient beyond the initial pilot scale.

The scalability assessment methodology (see Figure 26) follows a structured methodology implemented in the VLab:

1. **Define scenario scales:** Representative scaling levels are defined, such as small, medium, and large virtual energy communities (e.g., tens, hundreds, or thousands of prosumers). Each scale reflects a realistic deployment size that could be expected in future rollouts.

2. **Generate scalable configurations:** For each scale, the VLab generates configuration files describing the number of assets, communication topics, data models, and control interfaces. Asset replication is performed automatically using parameterized templates to ensure consistency across scales.

3. **Increase system complexity:** In addition to increasing size, operational complexity is introduced by adjusting messaging frequencies, control update intervals, and ICT characteristics such as data

volume and concurrency. This allows the evaluation of system behaviour under increasing computational and communication load.

4. **Execute experiments in the virtual environment:** The generated configurations are deployed and executed in the VLab. Virtual assets publish measurements and receive control commands according to the defined scenarios, while controllers interact with the system as in real operation.

5. **Monitor and evaluate KPIs:** Key ICT-related KPIs are continuously monitored, including message throughput, communication latency, CPU and memory usage, and the maximum number of supported assets or prosumers. These indicators are analysed to identify performance trends and scalability limits.

Through this methodology, the VLab provides quantitative and reproducible evidence of how the PARMENIDES solutions behave as system scale increases, supporting robust conclusions on scalability.

5.1.3 Replicability Assessment Methodology

The replicability analysis aims to demonstrate that the PARMENIDES solutions can be transferred to different technical and contextual settings with minimal engineering effort. Rather than focusing on system size, replicability addresses variation in system structure, asset composition, and operational context, such as residential, rural, or urban energy communities. The AIT VLab enables replicability analysis by supporting rapid reconfiguration of the virtual environment through standardized configuration templates. This approach ensures that the same solution stack can be deployed across multiple scenarios without changes to core software components, thereby demonstrating portability and adaptability. This is achieved through the definition of archetype sites representing typical energy system contexts, such as multi-apartment residential buildings, rural villages with high photovoltaic penetration, or urban electric vehicle charging hubs.

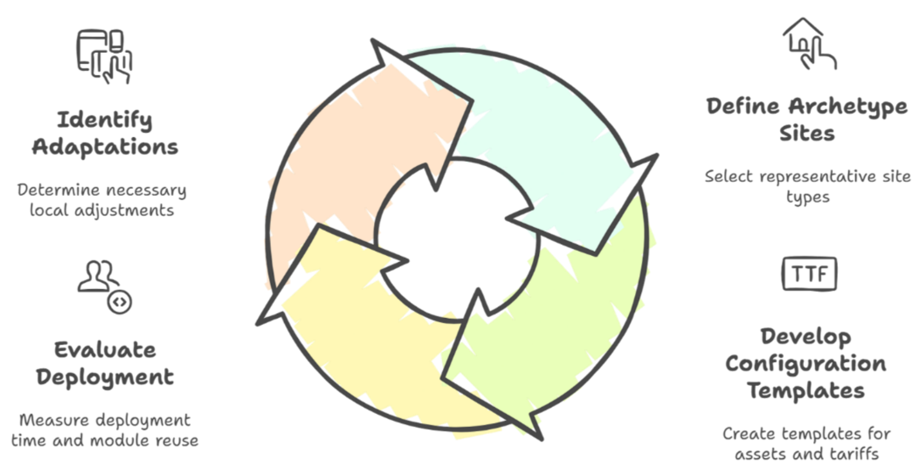


Figure 27: Overview of the AIT VLab-based replicability assessment methodology.

For each archetype, configuration templates are defined using standardized formats such as YAML or JSON. These templates describe assets, tariffs, communication topics, and control parameters. By switching between archetypes primarily through configuration changes rather than code modifications, the project demonstrates that PARMENIDES solutions are highly adaptable. Replicability is evaluated using KPIs such as deployment time, configuration effort, and functional consistency across archetypes. This approach provides concrete evidence that the solutions can be transferred to new sites and regulatory contexts with limited customization.

The replicability assessment in the AIT VLab (see Figure 27) follows a configuration-driven methodology:

1. **Define energy community archetypes:** Representative archetypes are identified, such as multi-apartment residential buildings, rural communities with high photovoltaic penetration, or urban areas with concentrated EV charging infrastructure. Each archetype reflects a distinct deployment context.

2. **Specify archetype configurations:** For each archetype, configuration templates are defined to describe assets, tariffs, communication topics, and control parameters. These configurations capture the structural differences between contexts while using the same underlying solution components.

3. **Deploy archetypes through configuration changes:** The VLab instantiates each archetype primarily through configuration changes (e.g., YAML or JSON files), without modifying application logic or controller implementations. This step demonstrates how easily the solution can be adapted to a new context.

4. **Execute scenario-based tests:** The same use case scenarios are executed across different archetypes to ensure functional consistency. Controllers and services interact with the virtual assets in the same way as in other configurations.

5. **Evaluate replicability KPIs:** Replicability is assessed using KPIs such as deployment time, configuration effort, and consistency of system behaviour across archetypes. Differences and deviations are analysed to identify context-specific considerations.

5.1.4 Controller-in-the-Loop (CIL) Testing

Controller-in-the-loop (CIL) testing is a central element of the Scalability and Replicability Analysis enabled by the AIT VLab. In this approach, real PARMENIDES controllers, such as EMS4HESS components, are connected directly to the virtual environment and interact with emulated assets using the same interfaces and communication mechanisms as in real deployments.

CIL testing bridges the gap between purely simulated studies and physical pilot operation. It allows control algorithms to be evaluated under realistic timing, data exchange, and operational conditions while maintaining full control over system configuration and experimental parameters.

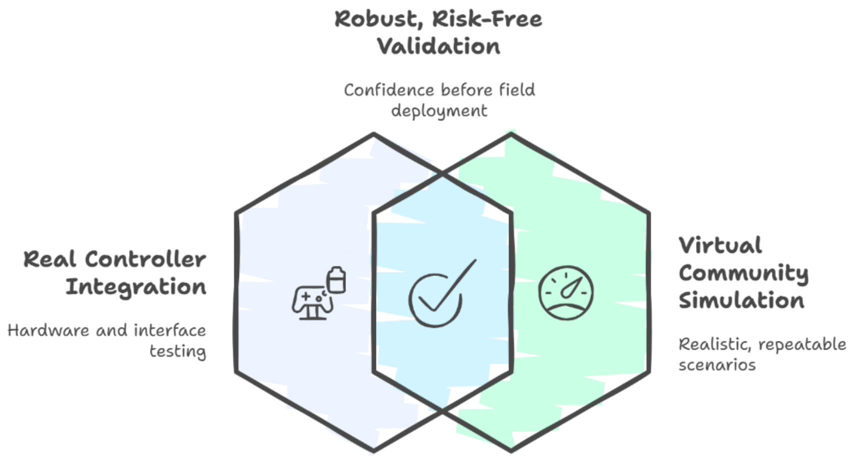


Figure 28: Conceptual overview of the provision of controller-in-the-loop testing for SRA enabled through AIT VLab.

The CIL testing process in the VLab follows a structured sequence:

1. **Integrate real controllers with the VLab:** Controllers are deployed alongside the VLab and connected through standard interfaces (e.g., REST/HTTP and MQTT). From the controller's perspective, the virtual assets are indistinguishable from real devices.

2. **Bind controllers to virtual assets:** Controllers are configured to subscribe to measurement data and issue control commands to the VLab-generated assets. Asset behaviour follows predefined profiles or time series to emulate realistic operation.

3. **Execute control scenarios:** Control strategies are executed under different system sizes, archetypes, and operating conditions. This includes normal operation as well as stress scenarios with increased load or communication intensity.

4. **Observe control behaviour and system response:** Controller responsiveness, stability, and interaction patterns are observed, including reaction times, convergence behaviour, and robustness against delayed or missing data.

5. **Evaluate control-related KPIs:** KPIs such as control latency, frequency of control actions, system stability indicators, and interaction with ICT constraints are analysed to assess controller performance under scalable and replicable conditions.

CIL testing enables the validation of control logic in scenarios that go beyond what can be safely or economically tested in physical pilots. It provides critical insights into how controllers behave under large-scale and diverse deployment conditions, thereby strengthening the overall SRA results.

5.1.5 SRA Scenarios and Results

Scalability and replicability tests were performed to evaluate how the EMS4HESS behaves when system size and data update frequency are increased. These tests aimed to observe runtime behaviour and execution characteristics under realistic constraints, rather than to determine theoretical maximum capacity.

All experiments were executed using the AIT VLab, which enabled repeated execution of the same configurations while systematically varying system size and timing parameters.

The test environment

EMS4HESS was deployed on a single physical server with an 8-core AMD processor running at 3.1 GHz and 16 GB of RAM. This hardware setup was chosen to reflect a deployment environment that could reasonably be expected in practice, without relying on cloud-based scaling or specialised high-performance infrastructure. A container-based deployment approach was used to structure the test environment. Separate containers were used to run EMS4HESS services, AIT VLab mock-ups representing pilot-specific controllers and devices, and components responsible for data ingestion and persistence. The database components were assigned their own share of computing resources so that data storage and dequeuing would not interfere with control execution during periods of increased load. This separation made it possible to observe the behaviour of control logic, data handling, and persistence processes individually, even though all components were running on the same physical machine.

The test scenarios

The tests covered three operating time resolutions:

1. A 15-minute interval was used to represent the standard community-level planning operations,
2. a 1-minute interval for more dynamic control behavior, and
3. a 1-second interval to stress the system under high-frequency data ingestion.

For each time resolution, the size of the energy community was gradually increased. Community composition was kept consistent across all scenarios to ensure comparability with the results. Prosumers represented 50% of the total members, and battery energy storage systems were assigned to 20% of the prosumers. These proportions were selected to reflect plausible community configurations rather than extreme or idealized cases.

The corresponding configurations were generated through the AIT VLab and executed without modifying the EMS4HESS core implementation. Virtual assets published measurement data and accepted control actions according to the selected time resolution. During execution, key performance indicators such as data ingestion latency, optimization execution time, and overall system stability were monitored.

The summary of results

Here is the summary of the results after conducting the test on the three scenarios:

- At a 15-minute time resolution, the system was able to handle configurations equivalent to approximately 30 000 community members on the single-server setup. At this scale, no stability issues were observed, and optimisation execution remained within acceptable time limits.
- More demanding tests were carried out using a 1-second ingestion interval. Under these conditions, the system successfully handled configurations with 40 active prosumers. The average ingestion latency measured during these runs was approximately 0.22 seconds.
- The average computation time required for community-level optimal planning in the high-frequency scenarios was approximately 2.4 seconds. This indicates that the optimisation routines

remain computationally efficient even when operating with frequent updates and limited execution windows.

Overall, the results show that EMS4HESS maintains stable behaviour across a wide range of system sizes and operating conditions. Achieving these results on a single, moderately provisioned server suggests that the system does not depend on excessive computing resources to scale. When combined with the configuration-driven setup enabled by the AIT VLab, the findings support the applicability of the solution to energy communities of varying size, composition, and operational requirements.

6 Challenges and Opportunities

6.1 Austrian pilot

In the Austrian pilot at the pilot sides Gasen and Heimschuh, the following challenges were encountered:

- In the course of the project the battery manufacturer which constructed the battery storage in Gasen became insolvent. This made the maintenance and the operation of the battery challenging since the availability of employees of the battery for support was at times quite limited.
- Some measurement devices were located on customers premises. This made maintenance difficult when physical access to the devices was needed and appointments with the customers had to be scheduled. A balance had to be found how often customers can be approached for physical access.
- A complex chain of systems interacted with each other in the pilots, some were deployed on customers premises, others on the premises of the DSO and further systems were hosted on infrastructure located at project partners. This made debugging challenging since multiple people from different project partners were involved.

In the course of the operation of the pilots, these opportunities were identified:

- Valuable data was collected in the pilots with a time granularity of ~½ minutes and stored in a time series database. This dataset is unique since data with such a high time resolution and measurement device density is rare for low voltage grids. This data can be used in future for virtual test beds were machine learned algorithms can be trained and their performance tested.
- Within the project, parts of the pilot were adjusted to be NIS2 ready. Since multiple project partners were involved. This was a good opportunity to identify and apply measures in a complex system to apply strict cybersecurity rules.
- During the course of the project the new Electricity Industry Act (Elektrizitätswirtschaftsgestz ElWG) came into force which grants, among, others, the possibility to have dynamic power limits for flexibilities. Valuable experiences could be collected with the Grid Capacity Management how such dynamic power limits can be calculated and applied in future.

6.2 Swedish pilot

In the course of the Swedish Pilot development and implementation, the following challenges were encountered:

- Full integration with EMS4HESS was not possible due to time constraints upon HESS Testbench completion, although implementing the Remote Node and the local PLC-based control system were significant steps forward
- Selecting an appropriate tool to add realism to the custom training environment required a shift from EnergyPlus to Modelica (Dymola), where a Modelica-based FMU proved most suitable
- Simulation models and the HESS Testbench were developed in parallel, with high-fidelity physical realism added later; nevertheless having both digital and physical infrastructure has been valuable

- To enable controlled testing and evaluation of HESS operation, involving human participants was not viable; however, this pivot provides insight for future deployment in energy communities

Moving forward, the following opportunities are seen:

- With both the physical and digital infrastructure in place, a trained controller can already be deployed and tested
- A machine-learned digital replica can also be developed in place of or together with a physics-based simulation in RL training of a control agent
- The control agent can later be trained to respond to dynamic temperature setpoints and baselines
- Transfer learning to systems with a different specification may be explored
- Realistic congestion forecasts and flexibility requests can be tested with the trained controller
- Currently, flexibility response is limited to intra-day activation; integrating day-ahead forecasts may show greater potential
- Aspects not fully exploited in the pilot, e.g., PECO, EMS4HESS, and VLAB can be further explored for application/integration

6.3 EMS4HESS

In the context of the project, MAPS EMS has been evolved to PARMENIDES EMS4HESS, an ontology-based EMS that, leveraging PECO, is able to manage diverse storage technologies and multiple energy vectors in a unified manner, under the HESS paradigm. Several challenges have been encountered along the project:

- **Semantic landscape fragmentation**: Despite EU initiatives like SAREF and consolidated power grid standards, the energy sector is hampered by proprietary vendor implementations and inconsistent data models. This fragmentation forces developers to absorb significant technical debt through the development of complex, bespoke semantic adapters. The lack of true "plug-and-play" interoperability significantly increases the cost and time required to onboard diverse energy assets, as we directly experienced in the deployment of the EMS4HESS.
- **Infrastructure and Data reliability gap**: Lack of clear business cases for investing in reliable and robust sensing and communication infrastructures at the MV/LV levels affects data quality and communication reliability. This also happened in our real-life pilots (e.g., no meters for PV generation, communication issues with some assets), partially limiting the effectiveness of some tests.
- **Closed-loop testing**: Transitioning from offline simulations to real-time, closed-loop control is technically demanding. It requires highly reliable testing environments to validate control logics without risking physical damages to the assets or impacts to the grid. This partially limited the number of experiments performed in the two Austrian pilots, while, as already mentioned, it wasn’t possible to close the loop in the Swedish pilot due to the unavailability of the HESS Testbench for full integration.

Opportunities:

- The Hybrid Energy Storage System modelling developed in PARMENIDES has already shown a high potential in supporting the implementation of new generation of multi-vector optimisation algorithms and techniques that can extend the capabilities of commercial EMS software, especially in the context of flexibility services, which is something that we want to further explore.

- Due to the fragmentation of ontologies and semantic interoperability approaches, we foresee the opportunity to explore modelling tools that separate the instantiation of energy systems (e.g., the configuration of our pilots) from their ontological representation, enabling independent evolution of system implementation and semantics, and facilitating the integration of diverse ontological models and semantic adaptation to specific use cases.

6.4 PECO

As Described in D5.1, PECO was used to describe pilot instances in Austria and Sweden. Specifically, the descriptions were defined for: energy community's and energy community members' details, equipment including energy supply terminals, sensors and actuators, equipment linkage (especially for heat) and prices and tariffs.

Throughout this effort, following challenges were identified that would benefit from addressing them in further work:

- Instantiation of pilots is currently performed by manually populating instantiation sheets and processing them using the developed scripts using the Protégé plugin Cellfie. The process, while relatively simple, is manual, error prone, and difficult to debug, and results are difficult to visualize and validate.
- While not feasible in PARMENIDES due to largely preexisting pilot infrastructure, a comprehensive semantic data interoperability layer would facilitate seamless data exchange and testing. At the moment this adaptation is happening at the EMS level.

Opportunities

- Ideally, semantic integration would be performed through connector plugins, and components would need to understand only PECO and other related ontologies (SAREF, SAREF4BLDG)
- Closer integration of VLAB testing framework models with other PECO modules would make semantic-sensitive testing possible as well
- Overall there is a need for an engineering tool facilitating simpler, automatic, or semi-automatic population of an instance based on PECO (or other) ontology, and allowing effective piecewise visualization and validation of generated instances.
- Another quality assurance improvement of the instantiation process could be gained by checking instantiated objects and relationships using SHACL-based (Shape Constraint Language) validation.

7 List of Figures

8 List of Tables

PARMENIDES

Plug&plAy eneRgy ManagEmeNt for hybrID Energy Storage