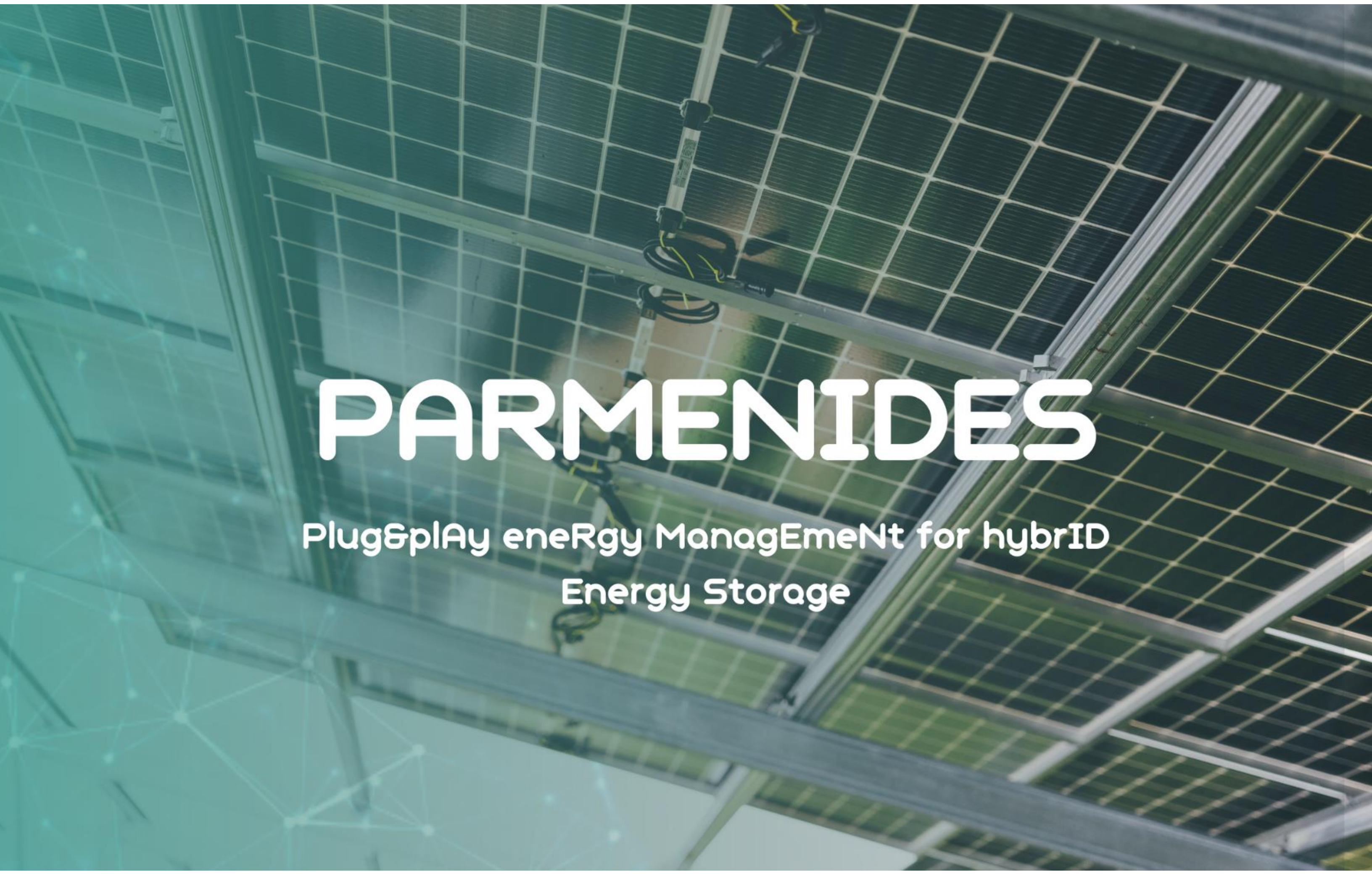

Deliverable D4.3

Virtual Verification Laboratory (VLab)

Work Package 4

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	WP4 – PARMENIDES System Development
Related task	T4.3 Provision of the virtual verification environment and integration testing
Deliverable due date	M20 (31.08.2024)
Actual delivery date	M24 (31.12.2024)
Dissemination level	PU – Public
Deliverable responsible	AIT

Document Information

Document Version: 1.0

Revision / Status: Submission

All Authors/Partners

Name	Organisation
Jawad Kazmi	AIT Austrian Institute of Technology GmbH
Mark Stefan	AIT Austrian Institute of Technology GmbH

Document History

Revision	Content/changes	Resp. partner	Date
0.1	Initial version	AIT	July 2024
0.2	Description of extensions	AIT	November 2024
0.3	Description of extensions	AIT	December 2024
1.0	Final version	AIT	17.12.2024

Document Approval

Final approval	Name	Resp. partner	Date
1.0	Mark Stefan	AIT	18.12.2024

Executive Summary

This document supports the PARMENIDES activities around integration and interoperability testing and contains a User Guide for AIT Virtual Lab (VLab). It will help you understand how to use the VLab framework to achieve high interoperability in PARMENIDES. The main Deliverable D4.3 is the VLab framework itself, this document serves as documentation.

Table of contents

Abbreviations

Acronym	Description
AI-flex	Autonomous AI for cellular energy systems increasing flexibilities provided by sector coupling and distributed storage
API	Application Programming Interface
ECOSINT	Energy COmmunity System INTegration
MQTT	Message Queuing Telemetry Transport
PECO	PARMENIDES Energy Community Ontology
PU	Public
REST	Representational State Transfer
SDL	Software Development Kit
SENDER	Sustainable Consumer engagement and demand response
SUT	System under Test
VLab	Virtual Verification Laboratory (long) / Virtual Lab (short)
VPN	Virtual Private Network
WP	Work Package

1. Introduction

Welcome to the AIT Virtual Lab (AIT VLab) User Guide. This guide will help you understand how to use the VLab framework to achieve high interoperability in your projects, in particular, in the PARMENDIES project. This document serves as documentation as part of the AIT Virtual Lab (AIT VLab) framework.

Interoperability is a key enabler of smart grid potential and should be regarded as an intrinsic component of any smart grid application being developed from its inception, see Figure 1. Because interoperability is a design consideration, thinking about it early on saves energy and resources. Conversely, when the engaged partners and stakeholders have a better understanding, consensus on, and knowledge of the automation interfaces, dependencies, and expectations the communication becomes more effective and easier.

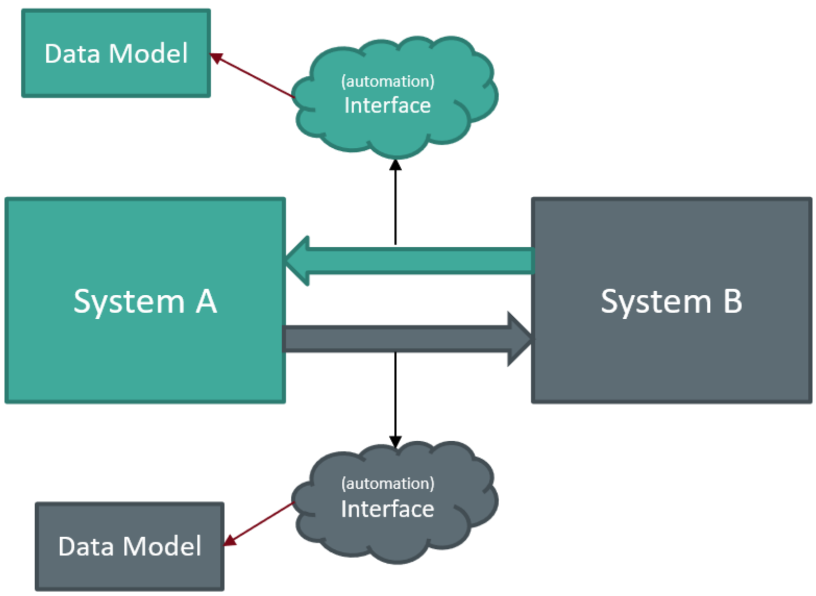

Figure 1: Interoperability between System A and System B

One of the challenges in achieving a high interoperability maturity level is having an incompatible and ad-hoc workflow. The AIT VLab addresses this challenge by providing a framework that includes a methodology and toolset for achieving a higher level (semantic and above) of interoperability. It advocates defining a common view of the system first so that the functional objectives of the solution can be aligned with what needs to be implemented. This way it also helps in bridging the knowledge and understanding gap between the requirement and implementation teams.

The framework is equally beneficial for system architects, developers, and most other stakeholders. The framework is an ecosystem of modules, interfaces, allowed operations, and the data model along with both synchronous (REST APIs and client SDKs) and asynchronous (publisher/subscriber model) architectures with accompanying documentation packed in a portable environment to provide a mockup prototype of the proposed system for testing and integration. In addition to generating the mock values, the generated mockups further can follow load/generation profiles to behave in a more deterministic and realistic manner. Such capabilities are used, for example, in doing scalability and replicability analysis.

The AIT VLab framework's ability to provide a comprehensive ecosystem for testing, integration, and scalability analysis makes it a versatile tool across these segments, helping bridge the gap between requirements and implementation.

AIT VLab stands out in the smart grid interoperability applications due to several unique features and advantages:

- **Comprehensive Framework with Integrated Methodology and Toolset**: AIT VLab provides a complete ecosystem of modules, interfaces, and data models, along with both synchronous (REST APIs and client SDKs) and asynchronous (publisher/subscriber model) architectures. This comprehensive approach ensures that all aspects of interoperability are covered, from design to implementation and testing.
- **Mockup Prototyping**: The framework includes capabilities for generating mock-values and following load/generation profiles, which allows for more deterministic and realistic testing scenarios. This is particularly useful for scalability and replicability analysis, ensuring that solutions are robust and reliable before deployment.
- **Ease of Use**:
  - **Portable Environment**: AIT VLab is packaged in a portable environment, making it easy to set up and use across different systems and projects. This reduces the time and effort required for integration and testing.
  - **User-Friendly Input Template**: The VLab Input Template simplifies the process of defining data models and specifying modules and interfaces, making it accessible to a wide range of users, from system architects to developers.
- **Proven Success**: AIT VLab has been successfully used in numerous research projects, including the Horizon 2020 project **SENDER**[1]**,** the actual Horizon Europe project **PARMENIDES,** the ERA-Net project **AI-Flex**[2]**,** and national projects such as **ECOSINT**[3], receiving very positive feedback. This track record demonstrates its effectiveness and reliability in real-world applications.
- **Alignment of Functional Objectives**: By advocating for a common view of the system from the outset, AIT VLab helps bridge the knowledge and understanding gap between requirement and implementation teams. This ensures that the functional objectives of the solution are aligned with what needs to be implemented, leading to more effective and efficient project outcomes.

---

[1] Horizon 2020 (#957755), Sustainable Consumer engagement and demand response (SENDER), https://www.sender-h2020.eu/

[2] ERA-Net SES, Autonomous AI for cellular energy systems increasing flexibilities provided by sector coupling and distributed storage (AI-flex)

[3] FFG Austria, Energy COmmunity System INTegration (ECOSINT), http://www.ecosint.at

2. Getting Started

To begin using VLab, you need to:

- **Download the VLab Input Template**: This template is an Excel file used to define your system's data models, modules, interfaces, operations, and parameters.
- **Install Required Software**: Ensure you have Microsoft Excel and Docker installed on your system.

3. Filling Out the Input Template

The AIT VLab currently supports the specification and generation of both the asynchronous and synchronous modules. The input templates for both these types have some slight differences.

The input template for specifying the Synchronous (RESTful) modules consists of five worksheets while the template for specifying the Asynchronous (publisher/subscriber) modules consists of only four sheets. Each worksheet serves a specific purpose in defining your system.

3.1. Data Modelling Worksheet (Sync/Async)

Purpose: Define the structure and properties of data models.

Steps:

- Enter the Data Model Name.
- Define **Member Names** and their properties (e.g., data type, default value, min/max values).

Data Model

Data Model Name	Member Name	Instance Name	Array of Objects	Description	Data Ty	Example Values	Default Value	Min. Value	Max. Value	Min. Leng	Max. Leng	Nullab	Enum	Pattern	DataMod	Unit
PhysicalValue	value		No	Value	Double	100	0,00	-999999999,00	999999999,00			No				
PhysicalValue	type		No	Type of Value	String							No				
StaticProperty	<PhysicalValue>	staticProperty	No		Custom							No				
Datapoint	<PhysicalValue>	dataPoint	No		Custom							No				
Measurement	<Datapoint>	measurement	No		Custom							No				
Setpoint	<Datapoint>	setpoint	No		Custom											
Timestamp	value		No		Long							No				
PowerSpec	<StaticProperty>	maxPowerCapacity	No	Maximum power capacity	Custom	20 000	0	0	999 999 999			No				W
PowerSpec	<StaticProperty>	maxPowerDensityVol	No	Maximum volumetric power density	Custom	5000	0	0	999 999			No				W/m^3
PowerSpec	<StaticProperty>	maxPowerDensityMass	No	Maximum specific power density	Custom	1000	0	0	999 999			No				W/kg
PowerParameter	<Measurement>	powerCapacity	No	Power capacity	Custom	5 000	0	0	999 999 999			No				W
PowerParameter	<Measurement>	powerDensityVol	No	Volumetric power density	Custom	5000	0	0	999 999			No				W/m^3
PowerParameter	<Measurement>	powerDensityMass	No	Specific power density	Custom	4,5	0	0	999 999			No				W/kg
PowerParameter	<Measurement>	power	No	Power as energy per unit time	Custom	5 000	0	-999 999 999	999 999 999			No				W
EnergySpec	<StaticProperty>	maxEnergyCapacity	No	Maximum energy capacity	Custom	5 000	0	0	999 999 999			No				kWh
EnergySpec	<StaticProperty>	maxEnergyDensityVol	No	Maximum volumetric energy density	Custom	1200	0	0	99 999			No				kWh/m^3
EnergySpec	<StaticProperty>	maxEnergyDensityMass	No	Maximum specific energy density	Custom	0,25	0	0	9			No				kWh/kg
EnergyParameter	<Measurement>	energyCapacity	No	Energy capacity	Custom	5 000	0	0	999 999 999			No				kWh
EnergyParameter	<Measurement>	energyDensityVol	No	Volumetric energy density	Custom	1200	0	0	99 999			No				kWh/m^3
EnergyParameter	<Measurement>	energyDensityMass	No	Specific energy density	Custom	0,25	0	0	9			No				kWh/kg
EnergyParameter	<Measurement>	energy	No	Usable energy content	Custom	5 000	0	0	999 999 999			No				kWh
ElectricalSpec	<StaticProperty>	electricalSpecsAC	No	Static electrical properties/specifications (alternating current)	Custom							No				
ElectricalSpec	<StaticProperty>	electricalSpecsDC	No	Static electrical properties/specifications (direct current)	Custom							No				
ElectricalSpec	<EnergySpec>	ratedEnergyCapElec	No	Rated electrical energy capacity	Custom	5 000	0	0	999 999 999			No				kWh
ElectricalSpecAC	<ElectricalSpec>	ratedPowerAC	No	Rated operating electrical power (AC)	Custom	5 000	0	0	999 999 999			No				W
ElectricalSpecAC	<ElectricalSpec>	maxPowerAC	No	Maximum electrical power (AC)	Custom	5 000	0	0	999 999 999			No				W
ElectricalSpecAC	<ElectricalSpec>	minPowerAC	No	Minimum electrical power (AC)	Custom	5 000	0	0	999 999 999			No				W
ElectricalSpecAC	<ElectricalSpec>	ratedCurrentAC	No	Rated operating current (AC)	Custom	10	0	0	999			No				A
ElectricalSpecAC	<ElectricalSpec>	maxCurrentAC	No	Maximum operating current (AC)	Custom	25	0	0	999			No				A
ElectricalSpecAC	<ElectricalSpec>	minCurrentAC	No	Minimum operating current (AC)	Custom	0	0	0	999			No				A
ElectricalSpecAC	<ElectricalSpec>	ratedVoltageAC	No	Rated operating voltage (AC)	Custom	230	0	0	999999			No				V
ElectricalSpecAC	<ElectricalSpec>	maxVoltageAC	No	Maximum operating voltage (AC)	Custom	240	0	0	999999			No				V
ElectricalSpecAC	<ElectricalSpec>	minVoltageAC	No	Minimum operating voltage (AC)	Custom	100	0	0	999999			No				V
ElectricalSpecAC	<ElectricalSpec>	ratedFrequency	No	Rated operating frequency	Custom	50, 60	50	50	60			No				Hz
ElectricalSpecAC	<ElectricalSpec>	maxFrequency	No	Maximum operating frequency	Custom	50.5, 60.3	50,5	50	61			No				Hz
ElectricalSpecAC	<ElectricalSpec>	minFrequency	No	Minimum operating frequency	Custom	49.5, 59.7	49,5	49	60			No				Hz
ElectricalSpecAC	<ElectricalSpec>	phaseConfig	No	Electrical phase configuration	Custom	3-phase, 1-phase						No				-
ElectricalSpecDC	<ElectricalSpec>	ratedPowerDC	No	Rated operating electrical power (DC)	Custom	5 000	0	0	999 999 999			No				W
ElectricalSpecDC	<ElectricalSpec>	maxPowerDC	No	Maximum electrical power (DC)	Custom	100	0	0	999 999 999			No				W
ElectricalSpecDC	<ElectricalSpec>	minPowerDC	No	Minimum electrical power (DC)	Custom	50	0	0	999 999 999			No				W
ElectricalSpecDC	<ElectricalSpec>	ratedCurrentDC	No	Rated operating current (DC)	Custom	5	0	0	999			No				A
ElectricalSpecDC	<ElectricalSpec>	maxCurrentDC	No	Maximum operating current (DC)	Custom							No				A
ElectricalSpecDC	<ElectricalSpec>	minCurrentDC	No	Minimum operating current (DC)	Custom							No				A
ElectricalSpecDC	<ElectricalSpec>	ratedVoltageDC	No	Rated operating voltage (DC)	Custom	24	0	0	999999			No				V
ElectricalSpecDC	<ElectricalSpec>	maxVoltageDC	No	Maximum operating voltage (DC)	Custom	48	0	0	999999	V		No				V
ElectricalSpecDC	<ElectricalSpec>	minVoltageDC	No	Minimum operating voltage (DC)	Custom	12	0	0	999999	V		No				V

Figure 2: VLab Input Template for Data Models.

3.2. Module Definition Worksheet (Sync/Async)

Purpose: Describe individual modules/components.

Steps:

- Enter the **Name** and **Description** of the module.
- Provide the **API Programmatic Name**, **Documentation Link**, **Version**, **Developing Partner**, and **Responsible Person**.

Modules

Name	Description	API Programatic Name	Documentation Link	Version	Developing Partner	Responsable Person
EcoFlow IoT Open Platform	Platform for the monitoring and control of the EcoFlow battery	ecoflow	https://parmenides-project.eu/	1.0.0	KTH	Lorenz Ray Payonga <payonga@kth.se>

Figure 3: VLab Input Template for Modules.

3.3. Interface Definition Worksheet (Sync only)

Purpose: Define interfaces for modules only in the case of RESTful or sync modules.

Steps:

- Select the Parent Module.
- Enter the Interface Name and Programmatic Name.
- Provide a brief Interface Description.

Interfaces

Parent Module	Interface Name	Programatic Nam	Interface Description
EcoFlow IoT Open Platform	Configure	config	The main confirguration interface.
EcoFlow IoT Open Platform	ElectricalParameters	elecparams	Real-time electrical parameter measurements
EcoFlow IoT Open Platform	StorageParameters	storageparams	Real-time storage parameter measurements
EcoFlow IoT Open Platform	ChargingParameters	chargingparams	Real-time charging parameter measurements
EcoFlow IoT Open Platform	DischargingParameters	dischargingparams	Real-time discharging parameter measurements
EcoFlow IoT Open Platform	ThermalParameters	thermparams	Real-time thermal parameter measurements
EcoFlow IoT Open Platform	Setpoints	setpoints	Interface for dynamic setpoints

Figure 4: VLab Input Template for Interfaces (for synchronous architectures).

PARMENIDES

3.4. Operations Definition Worksheet (Sync only)

Purpose: Specify operations (GET, SET) for interfaces.

Steps:

- Select the **Interface ID**.
- Choose the **Operation Type** (GET or SET).
- Provide a **Description** of the operation.

Operations		
InterfaceID	**Operation Type**	**Description**
EcoFlow IoT Open Platform:Configure	Get	Get the current configuration for this module.
EcoFlow IoT Open Platform:Configure	Set	Change current configuration for this module.
EcoFlow IoT Open Platform:ElectricalParameters	Get	Get the real-time measurements for this interface
EcoFlow IoT Open Platform:StorageParameters	Get	Get the real-time measurements for this interface
EcoFlow IoT Open Platform:ChargingParameters	Get	Get the real-time measurements for this interface
EcoFlow IoT Open Platform:DischargingParameters	Get	Get the real-time measurements for this interface
EcoFlow IoT Open Platform:ThermalParameters	Get	Get the real-time measurements for this interface
EcoFlow IoT Open Platform:Setpoints	Get	Get the current setpoints for this module
EcoFlow IoT Open Platform:Setpoints	Set	Change the current setpoints for this module

Figure 5: VLab Input Template for Operations (for synchronous architectures).

3.5. Parameters Assignment Worksheet (Sync only)

Purpose: Link data models to operations.

Steps:

- Select the **Operation ID**.
- Choose the **Parameter Data Model**.
- Enter the **Parameter Name** and **Description**.
- Specify the **Composition**, **Direction**, and whether it is **Required**.

Parameters						
OperationID	**Parameter Data Model**	**Parameter Name**	**Description**	**Composition**	**Direction**	**Required**
EcoFlow IoT Open Platform:Configure:Get	StaticProperty	currentConfig	Get the applicable configurations and specifications	Array	Output	Yes
EcoFlow IoT Open Platform:Configure:Set	StaticProperty	newConfig	Change applicable configurations and specifications	Array	Input	No
EcoFlow IoT Open Platform:ChargingParameters:Get	ChargingParameter	chargingParameters	Get charging parameters of interest	Array	Output	Yes
EcoFlow IoT Open Platform:DischargingParameters:Get	DischargingParameter	dischargingParameters	Get discharging parameters of interest	Array	Output	Yes
EcoFlow IoT Open Platform:StorageParameters:Get	StorageParameter	storageParameters	Get storage parameters of interest	Array	Output	Yes
EcoFlow IoT Open Platform:ElectricalParameters:Get	ElectricalParameter	electricalParameters	Get electrical parameters of interest	Array	Output	Yes
EcoFlow IoT Open Platform:ThermalParameters:Get	ThermalParameter	thermalParameters	Get thermal parameters of interest	Array	Output	Yes
EcoFlow IoT Open Platform:Setpoints:Get	Setpoint	currentSetpoints	Get the applicable setpoints	Array	Output	Yes
EcoFlow IoT Open Platform:Setpoints:Set	Setpoint	newSetpoints	Change applicable setpoints	Array	Input	No

Figure 6: VLab Input Template for Parameters (for synchronous architectures).

3.6. Channels Definition Worksheet (Async only)

Purpose: Define channels and link them to datamodels.

Steps:

- Choose the Parent Module.
- Enter a descriptive Channel Name (without any space or special character).
- Choose the Channel Message Data Model.
- Enter the Channel Message Name. It is possible to fill this with a "?", in which case it will be filled in with an autogenerated value.
- Enter a descriptive Channel Name.
- Enter Channel Address (topic for MQTT). The address can contain variable that can be substituted at runtime. They can be specified by enclosing them in {}, e.g. {DEVICE_ID}.
- Enter Channel Description with a short description of the defined channel.

Channels					
Parent Module	Channel Name	Channel Message Data Model	Channel Message Name	Channel Address	Channel Description
PAC2200	L1VoltageChannel	TimeValue	L1VoltgeMessage	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_Voltage/V	Measurement: Voltage of phase 1 (V)
PAC2200	r/L2_Voltage	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L2_Voltage/V	Measurement: Voltage of phase 2 (V)
PAC2200	r/L3_Voltage	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L3_Voltage/V	Measurement: Voltage of phase 3 (V)
PAC2200	L1_Current	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_Current/A	Measurement: Current of phase 1 (A)
PAC2200	r/L2_Current	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L2_Current/A	Measurement: Current of phase 2 (A)
PAC2200	r/L3_Current	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L3_Current/A	Measurement: Current of phase 3 (A)
PAC2200	r/L1_ActivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_ActivePower/kW	Measurement: Active power of phase 1 (kW)
PAC2200	r/L2_ActivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L2_ActivePower/kW	Measurement: Active power of phase 2 (kW)
PAC2200	r/L3_ActivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L3_ActivePower/kW	Measurement: Active power of phase 3 (kW)
PAC2200	r/L1_ReactivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_ReactivePower/kvar	Measurement: Reactive power of phase 1 (kW)
PAC2200	r/L2_ReactivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L2_ReactivePower/kvar	Measurement: Reactive power of phase 2 (kW)
PAC2200	r/L3_ReactivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L3_ReactivePower/kvar	Measurement: Reactive power of phase 3 (kW)
PAC2200	r/L1_ApparentPower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_ApparentPower/kVA	Measurement: Apparent power of phase 1 (kVA)
PAC2200	r/L2_ApparentPower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L2_ApparentPower/kVA	Measurement: Apparent power of phase 2 (kVA)
PAC2200	r/L3_ApparentPower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L3_ApparentPower/kVA	Measurement: Apparent power of phase 3 (kVA)
PAC2200	r/T0_ActivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T0_ActivePower/kW	Measurement: Total active power of all phases (kW)
PAC2200	r/T0_ReactivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T0_ReactivePower/kvar	Measurement: Total reactive power of all phases (kvar)
PAC2200	r/T0_ApparentPower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T0_ApparentPower/kVA	Measurement: Total apparent power of all phases (kVA)
PAC2200	r/T1_ActiveEnergyImport	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T1_ActiveEnergyImport/kWh	Measurement: Imported active energy (kW)
PAC2200	r/T1_ActiveEnergyExport	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T1_ActiveEnergyExport/kWh	Measurement: Exported active energy (kW)
PAC2200	r/T1_ReactiveEnergyImport	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T1_ReactiveEnergyImport/kvar	Measurement: Imported reactive energy (kvar)
PAC2200	r/T1_ReactiveEnergyExport	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T1_ReactiveEnergyExport/kvar	Measurement: Exported reactive energy (kvar)
PAC2200	r/T1_ApparentEnergy	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/T1_ApparentEnergy/kVAh	Measurement: Total apparent energy (kVAh)
energyLIVE	r/ActivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/energyLIVE/r/ActivePower/kW	Measurement: Active power (kW)
energyLIVE	r/ReactivePower	TimeValue	?	ENS/{PILOT}/{DEVICE_ID}/energyLIVE/r/ReactivePower/kvar	Measurement: Reactive power (kvar)
BatteryGasen	r/RealPower	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/RealPower/kW	Measurement: Active power (kW)
BatteryGasen	r/ReactivePower	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/ReactivePower/kvar	Measurement: Reactive power (kW)
BatteryGasen	r/SXSSOC	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/SXSSOC/	Measurement: State of Charge (%)
BatteryGasen	r/BatteryVoltage	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/BatteryVoltage/V	Measurement: DC voltage of the battery (V)
BatteryGasen	r/BatteryCurrent	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/BatteryCurrent/A	Measurement: DC current of the battery (A)
BatteryGasen	r/L1_Voltage	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L1_Voltage/V	Measurement: AC Voltage of phase 1 (V)
BatteryGasen	r/L2_Voltage	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L2_Voltage/V	Measurement: AC Voltage of phase 1 (V)
BatteryGasen	r/L3_Voltage	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L3_Voltage/V	Measurement: AC Voltage of phase 1 (V)
BatteryGasen	r/L1_Current	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L1_Current/A	Measurement: AC Current of phase 1 (A)
BatteryGasen	r/L2_Current	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L2_Current/A	Measurement: AC Current of phase 2 (A)
BatteryGasen	r/L3_Current	TimeValue	?	ENS/Gasen/Xelectrix/Battery/r/L3_Current/A	Measurement: AC Current of phase 3 (A)

Figure 7: VLab Input Template for Channels (for asynchronous architectures).

3.7. Operations Definition Worksheet (Async only)

Purpose: Link channels to operations.

Steps:

- Choose the **ChannelID**.
- Enter a descriptive **Operation Name**. It is possible to fill this with a “?”, in which case it will be filled in with an autogenerated value.
- Choose the **Action Type** to be enables on the selected channel. There are only two possible options; Send or Receive.
- Enter **Description** with a short description of the defined operation.

Operations			
ChannelID	**Operation Name**	**Action Type**	**Description**
PAC2200:L1VoltageChannel	?	Send	Module SENDS "L1VoltageMessage" message based on datamodel "TimeValue" over "channel 1" which maps to ENS/{PILOT}/{DEVICE_ID}/PAC2200/r/L1_Voltage/V on the broker/middleware
BatteryGasen:r/BatteryCurrent	?	Send	
BatteryGasen:r/BatteryVoltage	?	Send	
BatteryGasen:r/L1_Current	?	Send	
BatteryGasen:r/L1_Voltage	?	Send	
BatteryGasen:r/L2_Current	?	Send	
BatteryGasen:r/L2_Voltage	?	Send	
BatteryGasen:r/L3_Current	?	Send	
BatteryGasen:r/L3_Voltage	?	Send	
BatteryGasen:r/ReactivePower	?	Send	
BatteryGasen:r/RealPower	?	Send	
BatteryGasen:r/SXSSOC	?	Send	
BatteryGasen:w/RealPowerReq_AIT	?	Send	
EMS4HEMS:c/ActivePower	?	Send	
energyLIVE:r/ActivePower	?	Send	
energyLIVE:r/ReactivePower	?	Send	
GCM:c/max(ActivePower)	?	Send	
GCM:c/min(ActivePower)	?	Send	

Figure 8: VLab Input Template for Operations (for asynchronous architectures).

4. Using AIT VLab's Environments

VLab provides two types of virtual environments: VLab Portable and VLab Central.

4.1. VLab Portable

Purpose: Local testing and development environment.

Steps:

- **Download VLab Portable**: Obtain the Docker images and scripts from the AIT repository.
- **Run VLab Portable**: Use the provided scripts to set up and run the environment on your local machine.
- **Develop and Test**: Integrate and test your modules within the local environment.

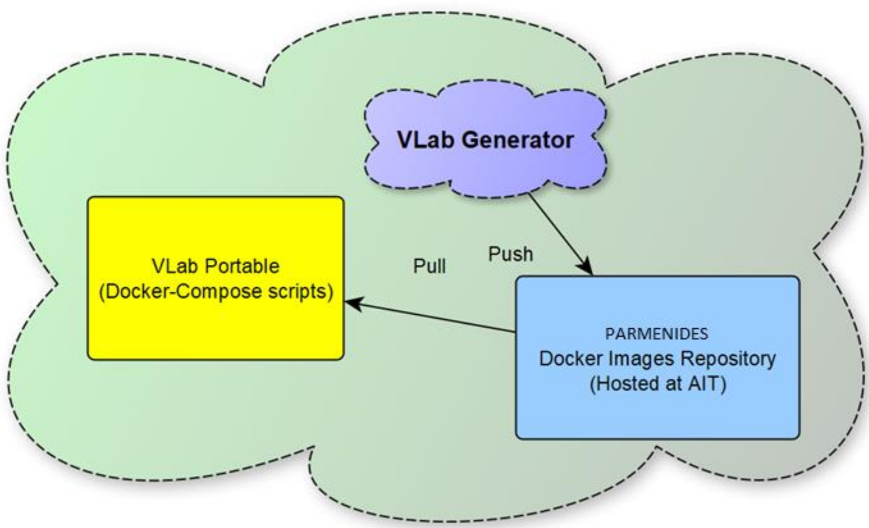

Figure 9: Overview of VLab Portable (yellow), based on generated Images (blue) using the VLab Generator (violet).

4.2. VLab Central

Purpose: Shared environment for system integration and testing.

Steps:

- **Access VLab Central**: Obtain VPN credentials from AIT.
- **Deploy Modules**: Upload your developed modules to the central repository.
- **Integration Testing**: Connect and test your modules with those developed by other partners.

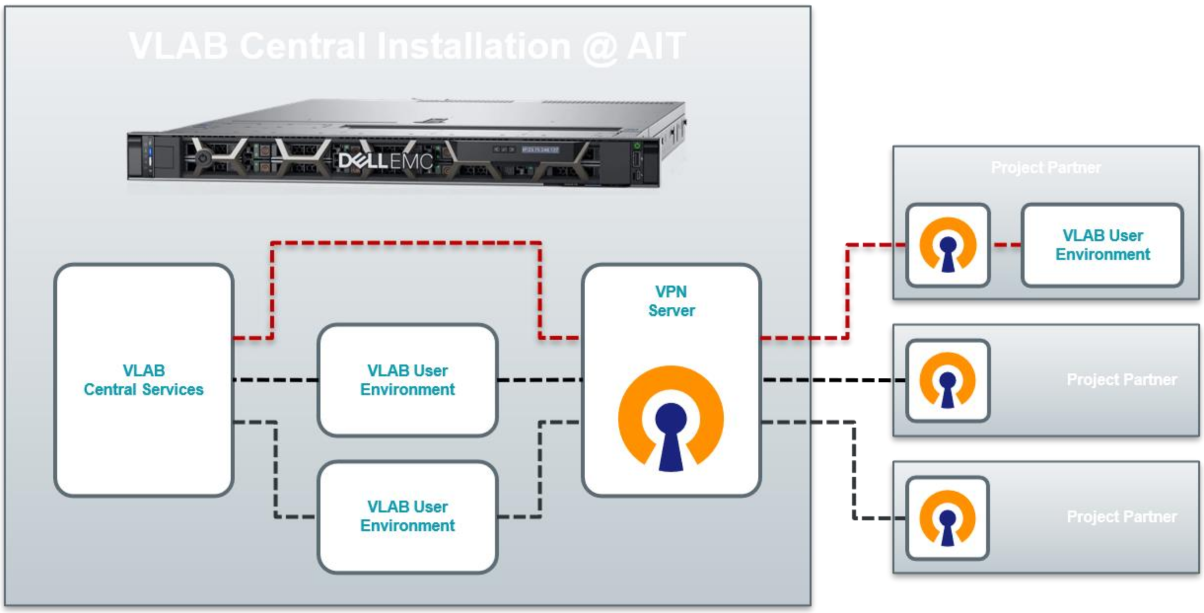

Figure 10: Overview of VLab Central, hosted at AIT with access to project partners for integrating their modules.

5. AIT VLab Validator

Purpose: Provides the capabilities to compliance test an implementation of a server or a client.

Steps: Follow the schematic show below:

- **Obtain the Validator Module**: Each (sync) module generated with AIT VLab will have a corresponding validator module automatically generated. First obtain the container image for your system-under-test.
- **Run System-Under-Test**: Run the (implementation) of your system-under-test.
- **Run your client module**: Now run your test code from a client module by sending and receiving messages to the SUT. You will have to adjust your client code to talk to the validator as proxy for the SUT by changing to the configured port. This would mean that you will not talk directly to SUT but call all the function using the validator instead.
- **See Results**: The validator module provides a Web UI can present a report highlighting all the exchanges and can highlight any non-conformity issues.

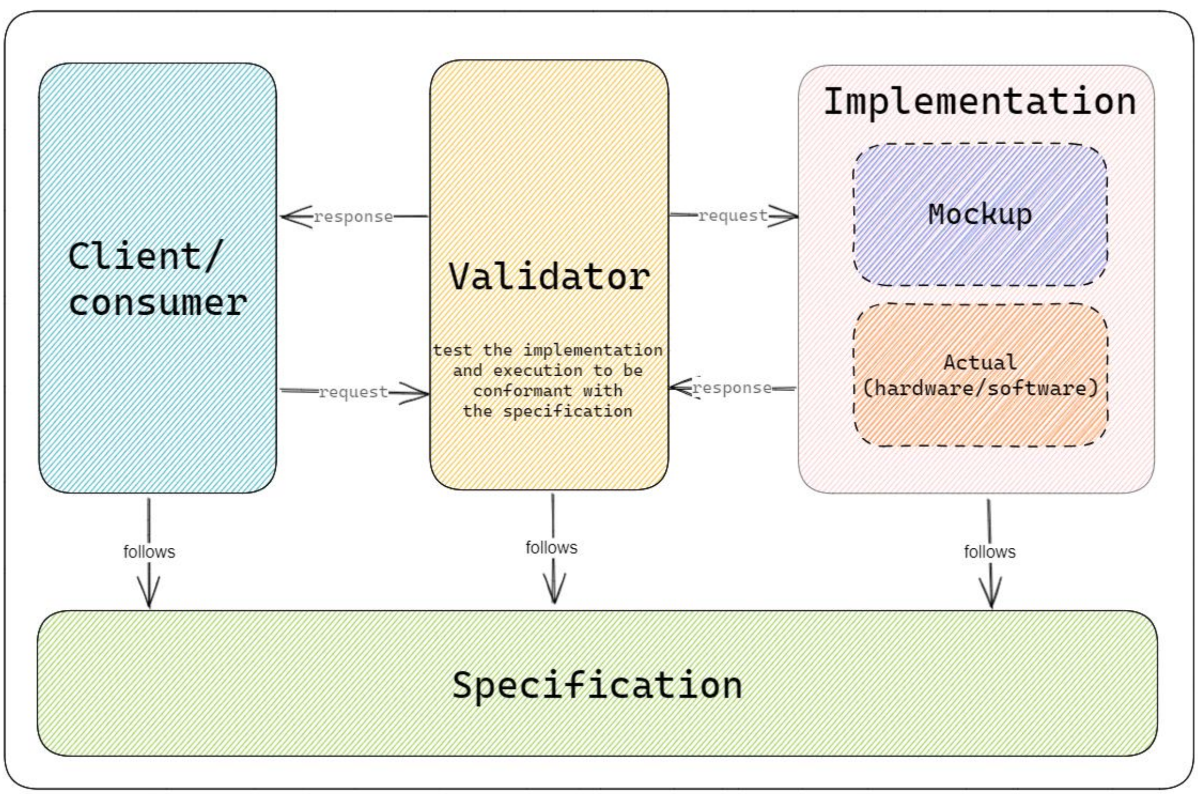

Figure 11: Overview about the Validator functionality.

6. Using power profiles with AIT VLab

Purpose: To use a profile with deterministic values instead of using the standard VLab mockups with random values.

Steps:

- **Obtain the profiles-enabled Module**: Obtain the required container image for the module you like to run.
- **Configure your module**: Make the configuration if the default behaviour is not suitable. The following are the most important environment variable to know about:
  - `PROFILE_FILE` (Default: "h0"): the source of the profile, there are 27 different types of profiles embedded in the VLab mockup for the year 2024, that can be switch by changing this variable. For more information check https://www.apcs.at/de/clearing/technisches-clearing/lastprofile.
  - `PROFILE_INTERVAL` (Default: 15 minutes): you can "simulate" how the timestamp changes for each new value read.
  - `PROFILE_SF` (Default: 1): The profile scaling factor value configured with this variable will be multiplied to each profile value read.
  - `PROFILE_START_TS` (Default: current data & time): Using this variable, the start timestamp can be provided that will then be used with each new profile value read.
  - PROFILE_START_INDEX (Default: 0): The starting index from where the profile should be read.
  - PROFILE_END_INDEX (Default: profile values count): The last value to be read. After the last value, the profile is roll-overed.

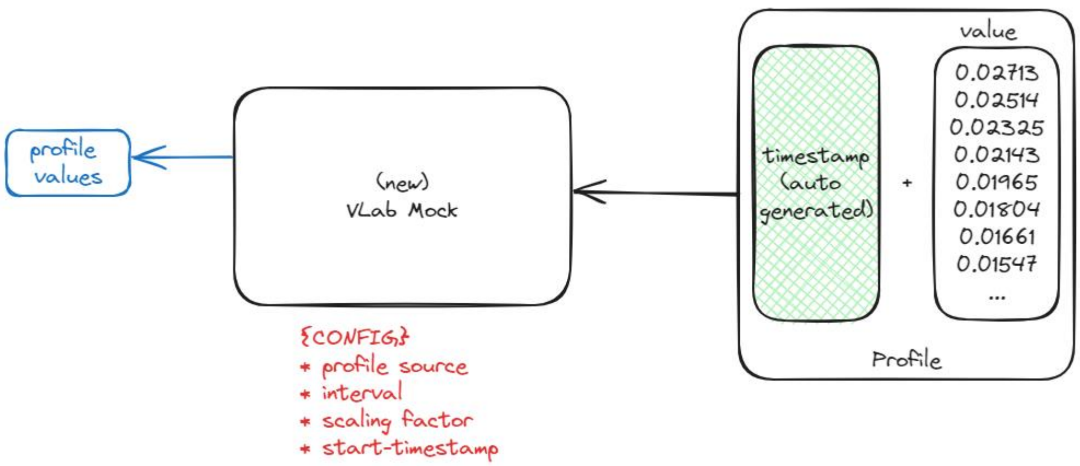

Figure 12: Using power profiles instead of random values in VLab Mockups.

7. Troubleshooting and Support

- **Common Issues:**
  - **Template Errors**: Ensure all required fields are filled out correctly.
  - **Docker Issues**: Verify Docker is installed and running properly.
- **Support**: Contact the responsible person listed in the module definition for specific module-related queries. For general support, refer to the AIT support team.

8. Annex

8.1. List of Figures

PARMENIDES


PARMENIDES
Plug&plAy eneRgy ManagEmeNt for hybrID
Energy Storage