SMART GRID SERVICES IMPLEMENTATION AND LAB-TESTING

D4.3: Smart grid services implementation and lab-testing

Date: **31/03/2022**
Author(s): **Alberto Zambrano**
Co-author(s): **María Tomás, Sílvia Valero, Laura Bagès, Katharina Csillak, Richard Kemmerzehl, Kinga Lőcsei-Tóth, Maria Chiara Cavarretta, Jussi Saari**

Deliverable details

Project number	Project acronym	Project title
875187	USER-CHI	Innovative solutions for USER centric Charging Infrastructure

Title	WP	Version
D4.3: Smart grid services implementation and lab-testing	WP4	1.0

Contractual delivery date	Actual delivery date	Delivery type*
M26 (March 2022)	M26 (March 2022)	R

*Delivery type: **R**: Document, report; **DEM**: Demonstrator, pilot, prototype; **DEC**: Websites, patent fillings, videos, etc; **OTHER**; **ETHICS**: Ethics requirement; **ORDP**: Open Research Data Pilot.

Author(s)	Organisation
Alberto Zambrano, María Tomás	ETRA
Sílvia Valero	AMB
Laura Bagès	CIR
Katharina Csillak	IKEM
Richard Kemmerzehl	GEW
Kinga Lőcsei-Tóth	BUD
Maria Chiara Cavarretta	ENEL
Jussi Saari	TUR

Document history

Version	Date	Person	Action	Status*	Dissemination level**
0.1	21/01/2022	Alberto Zambrano (ETRA)	ToC	Draft	CO
0.2	10/03/2022	Alberto Zambrano (ETRA) Katharina Csillak (IKEM) Richard Kemmerzehl (GEW) Kinga Lőcsei-Tóth (BUD) Maria Chiara Cavarretta (ENEL) Jussi Saari (TUR)	First draft	Draft	CO
0.3	21/03/2022	Alberto Zambrano, María Tomás (ETRA) Sílvia Valero (AMB) Laura Bagès (CIR)	Second draft	Draft	CO
0.4	24/03/2022	Laura Bagès (CIR) Natascia Andrenacci (ENEA) Holger Rapp, Lars Balzer (QWE)	Peer review	Draft	CO
1.0	31/03/2022	Alberto Zambrano, María Tomás (ETRA)	Final edit	Final	PU

*Status: Draft, Final, Approved, Submitted (to European Commission).

Dissemination Level: **PU: Public; **CO**: Confidential, only for members of the consortium (including the Commission Services); **EU-RES** Classified Information - restraint UE; **EU-CON**: Classified Information - confidential UE; **EU-SEC**: Classified Information - secret UE

Abstract

The main objective of the SMAC service is to provide CPOs the ability to implement smart charging strategies over the Charging Points under their control, giving them the possibility to outsource the calculation of those optimum profiles to a third-party service provider (the Smart Charging Service Provider actor - SCSP). This approach will help them to optimize their energy-related costs, enabling a better utilization of renewable energy sources and allowing their participation as active actors in the smart grid management, both as participants of implicit demand-side management strategies (i.e. by their exposure to dynamic energy tariffs) and explicit demand-side management campaigns (i.e. by being required to temporary reduce their load to support the grid management).

Keywords

Smart Charging, Smart Grid, technical, architecture, features, SMAC, development, testing, documentation

Copyright statement

Executive summary

The current deliverable, D4.3 "Smart grid services implementation and lab-testing", provides an overview of the work performed in the context of tasks T4.3 "Smart grid integration module development" and T4.4 "Smart grid services lab-testing and refinement".

These tasks focus in the development and testing of SMAC. The main objective of the SMAC service is to provide CPOs the ability to implement smart charging strategies over the Charging Points under their control, giving them the possibility to outsource the calculation of those optimum profiles to a third-party service provider (the Smart Charging Service Provider actor - SCSP). This approach will help them to optimize their energy-related costs, enable a better utilization of renewable energy sources and allow their participation as active actors in the smart grid management, both as participants of implicit demand-side management strategies (i.e. by the exposure to dynamic energy tariffs) and explicit demand-side management campaigns (i.e. by being required to temporary reduce their load to support the grid management).

These features are supported by a set of software components that interoperate to enable all the required services. Technical details and end-user documentation of these software components are listed in the content of the deliverable.
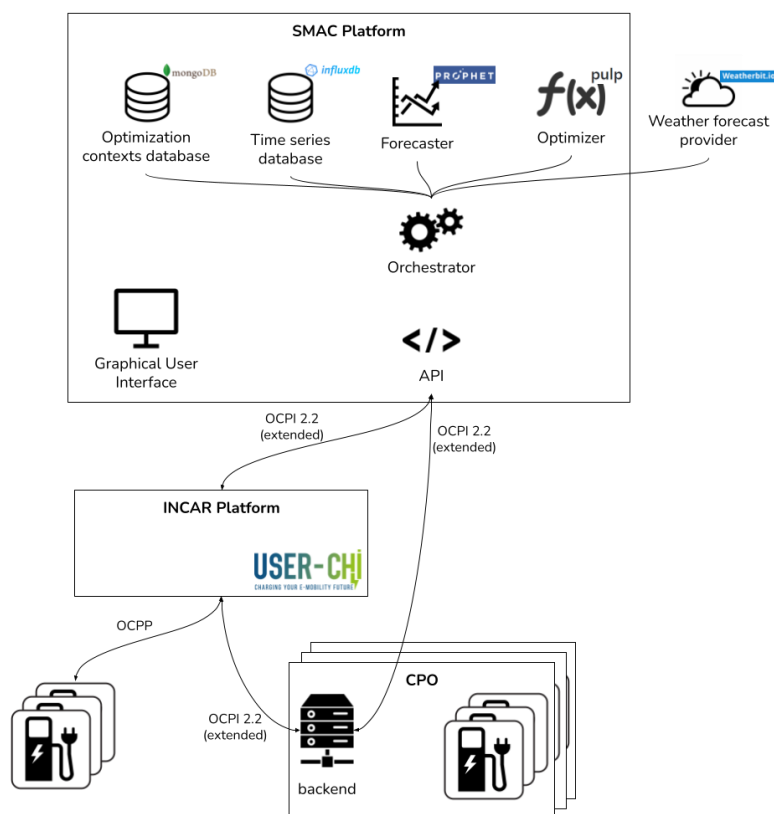
In order to formalize the testing phase, a testing framework has been proposed to conduct the lab-testing phase in a consistent and properly documented manner. Tests have been formally documented using a common template that compiles relevant information. Motivated by the primary objective at the current stage of the USER-CHI project within WP8, which is the preparation of the upcoming piloting phase, tests performed and compiled in this document have a strong focus on functionality.

Finally, this document provides some details on the effort taken so far by the pilot sites (Barcelona, Berlin, Budapest, Rome and Turku) with regards to the preparation of the pilots to successfully demonstrate the SMAC service in the upcoming months.

Table of contents

List of figures

List of tables

1. Introduction

1.1 Purpose of the document

The purpose of this deliverable is to document the work carried out in tasks T4.3 "Smart grid integration module development" and T4.4 "Smart grid services lab-testing and refinement". These tasks take over the work of T4.2 "Smart grid services design" in order to implement the required software solution, following the design guidelines documented in D4.2 "Smart grid services design" [1].

1.2 Scope of the document

This document has the objective of describing the implementation of SMAC, a service aiming to provide CPOs the ability to implement smart charging strategies over the Charging Points under their control, in order to optimize their energy-related costs, enable a better utilization of renewable energy sources and allow their participation as active actors in the smart grid management, both as participants of implicit demand-side management strategies (i.e. by the exposure to dynamic energy tariffs) and explicit demand-side management campaigns (i.e. by being required to temporary reduce their load to support the grid management).
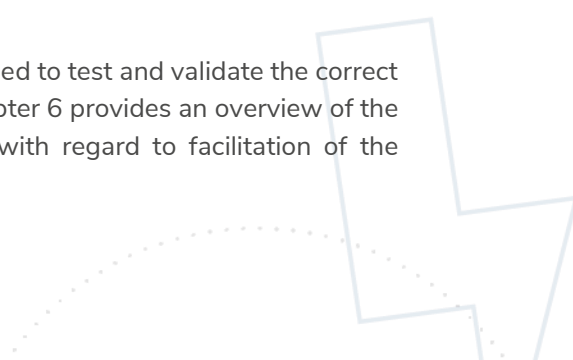
Due to the variety of targeted readers, sufficient details are provided from different perspectives, including overviews of the implemented features, technical details useful to CPOs with interest in integrating their backends with SMAC, and end-users utilizing the user interfaces to access to the information provided by SMAC.

Additionally, the document provides details on the efforts taken so far by the different demo sites of the USER-CHI project in the direction of facilitating the demonstration of SMAC.

1.3 Structure of the document

This document consists of seven chapters, including chapter 1 for the introduction to the deliverable and chapter 7 for the conclusions drawn during its writing. The core of the deliverable is contained in chapters 2 to 4, which provide the main details of the work performed. Chapter 3 includes an overview of SMAC features. Chapter 4 provides technical details about the architecture of the solution. Chapter 5 includes documentation of SMAC, of interest to parties willing to integrate their backends with SMAC.

Additionally, chapter 5 provides details on the methodology used to test and validate the correct operation of the provided software implementation, while chapter 6 provides an overview of the efforts taken by the Demo Sites of the USER-CHI project with regard to facilitation of the demonstration of SMAC in the corresponding sites.

2. SMAC features overview

2.1 Smart Charging as a Service

The main objective of the SMAC service is to provide CPOs the ability to implement smart charging strategies over the Charging Points under their control, giving them the possibility to outsource the calculation of those optimum profiles to a third-party service provider (the Smart Charging Service Provider actor - SCSP).

Its primary target is therefore to facilitate the access of any CPO to this feature. In order to do so, SMAC takes advantage of the overall framework constructed in the USERCHI project, and defines itself as a service fully integrated with the INCAR Platform. This facilitates the usage of the service by any CPO adhered to the INCAR Platform.

Additionally, SMAC implements – and extends – the required OCPI 2.2 [2] interfaces, and therefore can be used by any CPOs implementing such protocol, even if they are not registered in the INCAR Platform.

Figure 1 Integration options of SMAC
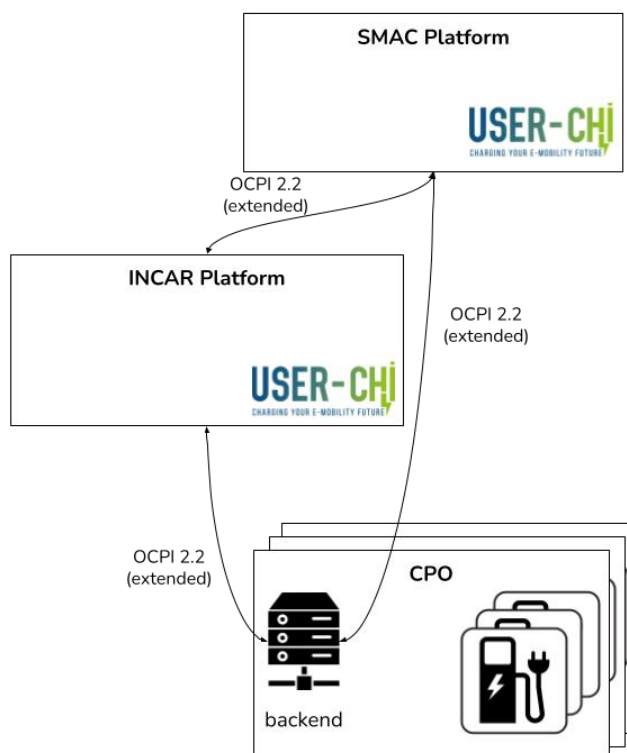
2.2 Infrastructure-level optimization

SMAC optimization algorithms take into account the complex reality of Charge Point managers, who, in general operate the Charge Points under a supply point that is shared with additional elements (typified as unmanageable loads, renewable generation – RES -, or local stationary energy storage). In these scenarios, Charge Points shall not be managed independently, since most energy management benefit come when those are considered as one piece of the complete picture. Nevertheless, the optimization model implemented within SMAC is versatile in the sense that is covers several combinations of field assets, going from more simple scenarios (e.g., load balancing among Charge Points sharing a supply point) to complex scenarios with consideration of local generation, storage and unmanageable loads.

Figure 2 Elements considered in the optimization model of SMAC
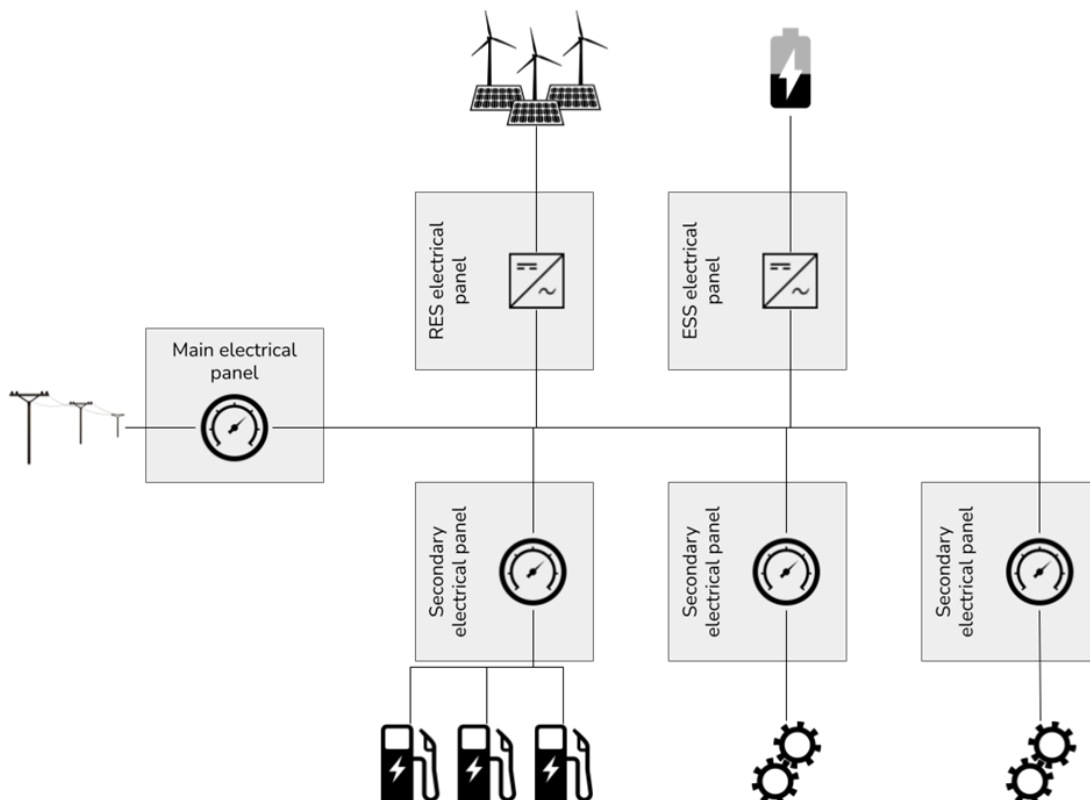
2.3 Flexibility services to grid operators

SMAC facilitates CPOs participation in flexible markets, allowing them to become active actors in the smart grid management by providing flexibility services to third party grid operators. Two main mechanisms are considered to enable CPOs as active actors in the smart grid management:

- Participation on implicit demand-side management strategies. These strategies usually modulate the demand of the participants by sending appropriate price signals (e.g. by offering a dynamic energy tariff scheme to the participants). The cost-wise optimization nature of SMAC inherently takes into account these signals to schedule the demand in the most cost-effective time slots

- Participation in explicit demand-side management strategies. These strategies usually consist on temporary constraints – in terms of maximum active power supplied – that need to be met by participants in order to support a proper smart grid operation. CPOs can communicate the requirements imposed by the corresponding grid operators to SMAC, which will consider it in the calculation of the charging profiles, thus ensuring that imposed temporal constraints over the power supply are met.

Figure 3 Flexibility request and its effect in the charging profile. First chart displays supply point capacity limits, including a temporal reduction of the available capacity triggered by the grid operator through a flexibility order. Second and third charts display the resulting smart charging profile and expected state of charge respectively. The effect of the flexibility order van be observed in the resulting profile
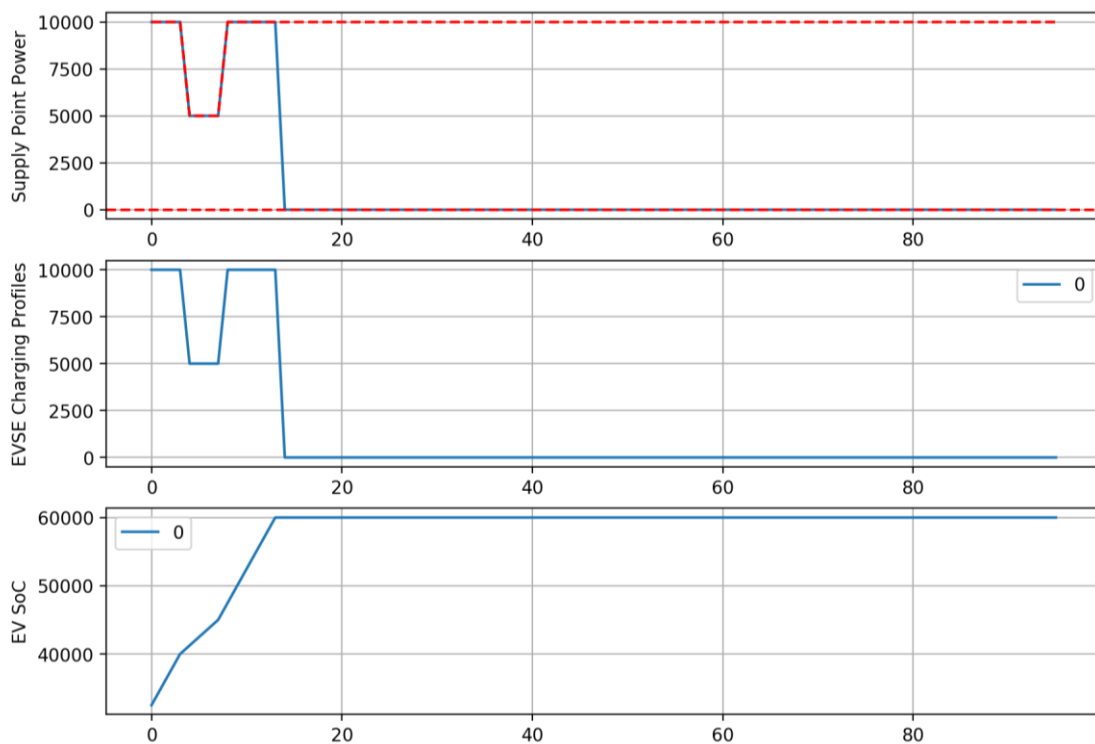
3. Architecture overview

3.1 SMAC component overview

In order to implement all features described in section 2, a number of components have been designed and implemented, each specialized for a particular task.

Figure 4 SMAC architecture overview
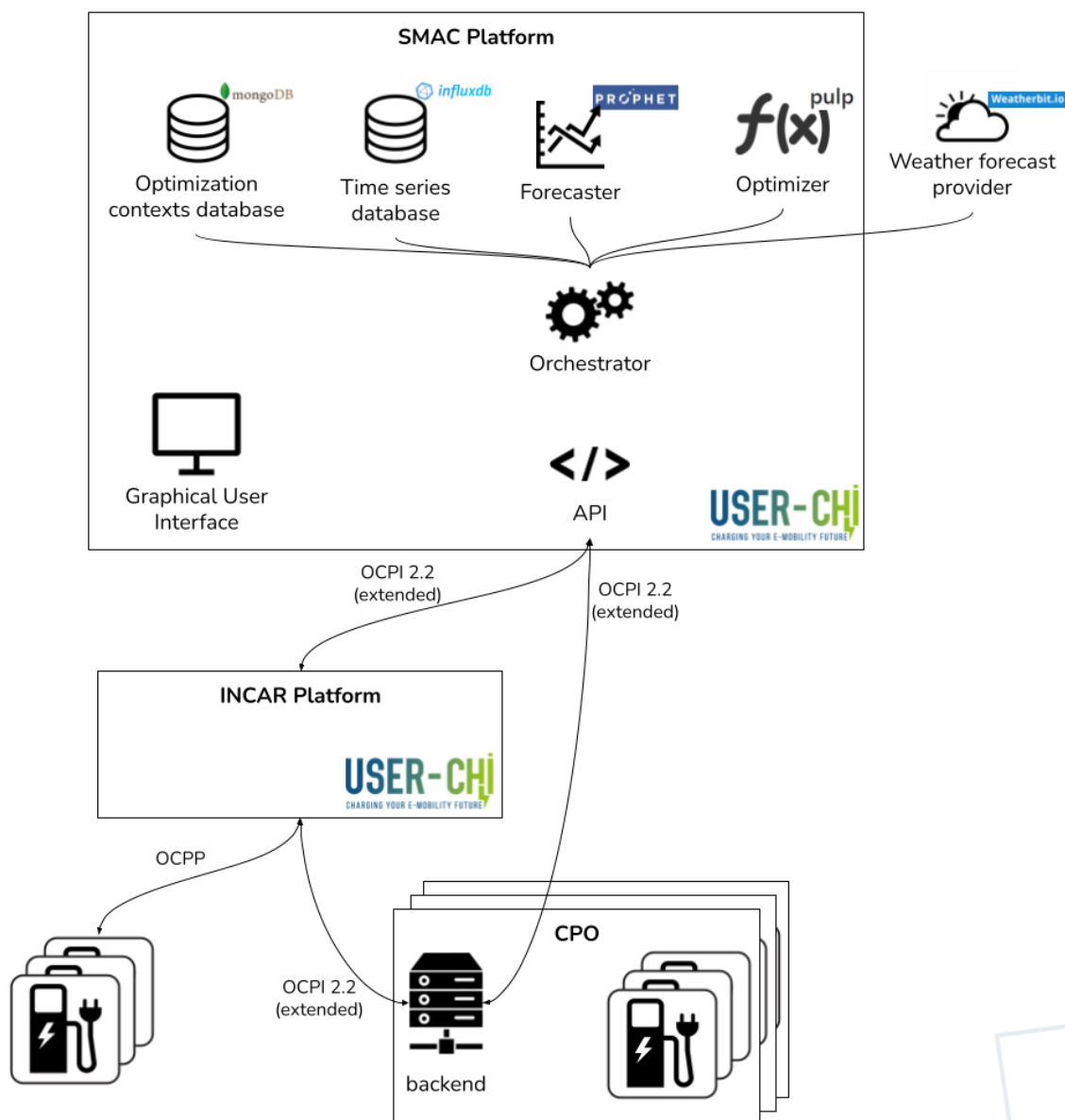
SMAC is composed by the following components:

- API: implements the required interfaces of OCPI 2.2 for the Smart Charging Service Provider role. Additionally, OCPI specification has been extended to allow CPOs to provide additional inputs to be considered in the calculation of the optimum profiles, as documented in section 4.1.

- Orchestrator: retrieves information from the different sources and calls the different internal services accordingly to the requirements, triggered by the API

- Optimization contexts database: stores the properties of the optimization contexts configured by the end users (CPOs)

- Time series database: stores historical data provided by the end users (demand, generation, energy prices, flexibility requests...)

- Forecaster: provides forecast of demand and generation time series, as described in section 3.2.5

- Optimizer: implements the linear optimization model in charge of calculating optimum charging profiles both for EVSEs and local storage (if available), as described in section 3.2.6

- Weather forecast provider: provides weather data, including actual and forecasted reports, which is used by the forecaster to build the generation forecast models

- User interface: allowing CPOs to configure the parameters of SMAC

The following sections provide further details on each one of those components

3.2 Components

3.2.1 API

The API component consists of an implementation of the OCPI 2.2 interface (with the defined extensions) built with Express for Node.js[1]. Details of the implemented API are provided in section 4.1.

3.2.2 Orchestrator

The orchestrator component has been implemented for Node.js[2]. This particular component has connections with the rest of the components of SMAC, and performs the following tasks:

---

[1] https://expressjs.com/

[2] https://nodejs.org/es/

- Registers or updates the definition of the *Optimization Contexts* in the corresponding database. Information about the changes to be registered may be provided by the *User Interface* via the extension of the OCPI 2.2 API.

- Whenever CPO provides new data about local demand, local generation, forecasts, energy prices or flexibility requests, the orchestrator component registers this information in the *Time series database*

- Whenever CPO registers data about local demand or generation for the first time, the orchestrator triggers the *Forecaster component* in order to create a new model for those metrics

- Upon reception of any update on any of the inputs of the smart charging profiles calculation (updates on any related *Session* object, or registration of new inputs from the CPO, including load or generation readouts, updates on energy price or registration of new flexibility requests), the following processes are triggered:

  o Update of the affected load or generation forecasts

  o Calculation of up-to-date smart charging profiles

  o Submission of update smart charging profiles to the corresponding CPO backend

- Periodically, the weather service provider is triggered in order to retrieve the information required by registered generation forecast models (current weather data and next 72 hours weather forecast)

3.2.3 Optimization contexts database

The optimization contexts database stores the static information about the environments for which optimum smart charging profiles will be calculated. An optimization context consists on:

- Related CPO, identified by its country code and party id (as per OCPI 2.2)

- Supply point constraints, including maximum values for supplied and injected active power, according to the contract between CPO and the corresponding grid operator (usually, the Distribution System Operator – DSO)

- Value for the *opportunity cost* synthetic cost. This parameter models the cost faced by the CPO not being able to give service to new drivers while the charge point is occupied, and is used internally by the optimization algorithm to discourage long-lasting charging sessions. More details are provided in section 3.2.6

- List of Charging Points to be considered together in the optimization (i.e., sharing the same supply point)

MongoDB[3] software is used to manage and store this information.

3.2.4 Time series database

The time series database is used to store all the dynamic inputs provided by the CPOs that need to be considered in the calculation of the optimum smart charging profiles. These dynamic inputs will be of one of the following types:

- Local unmanaged demand: additional demand sharing the same supply point with the charging points

- Local RES generation: RES source injecting energy under the same supply point where the charging points are located

- Local unmanaged demand/RES generation forecasts

- Energy prices, according to the conditions of the energy supply contract established between CPO and retailer

Additionally, all charging profiles calculated by SMAC are also persisted in this database.

InfluxDB[4] software is used to persist all this information.

Figure 5 Charging profiles stored by time series database
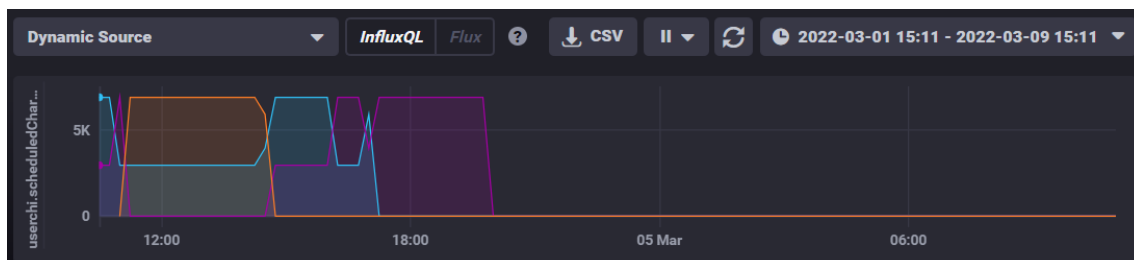
3.2.5 Forecaster

In order to calculate the optimum smart charging profiles for the charging points in those contexts that include local unmanaged demand and/or RES generation, the optimization model requires knowing the forecasted values for this demand and generation over the next 24 hours. In order to facilitate the adoption of SMAC as third-party Smart Charging Service Provider, it includes a forecaster module that will be automatically used to calculate the forecasts in those cases where the CPO is not providing such information.

---

[3] https://www.mongodb.com/es
[4] https://www.influxdata.com/

The forecaster component is built upon the Prophet library [3], which has raised a lot of interest the recent years. According to the authors of the library, "Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well".

Selection of the library has been made taking into account that several of its core characteristics make it appear as an interesting candidate, namely:

- The data to be forecast (demand and generation) meet the seasonality requisites for the time series required by the methods applied by the library
- The library handles the existence of gaps in the historical data, which facilitates its integration in the technical solutions from an operational perspective (e.g., no specific treatment to the input in order to cover data gaps or irregular reporting periods is required)
- Outliers are also automatically handled by the methods implemented by the library, which simplifies the required pre-processing steps on the input data

As detailed in *Forecasting at scale* [4], the techniques utilized include a decomposable time series model with three components: trend, seasonality – with different periods ranging from yearly to daily seasonality – and holidays (this latter one not applicable to the generation model presented hereby). Additionally, exogenous variables can also be taken into account. Main characteristics can be summarized as follows:

- Trend component may be modelled using two models internally: a *non-linear saturating growth* model and a *linear trend with changepoints* model
- Seasonality gets modelled by a set of *Fourier series*
- Exogenous variables (weather-depending component in the case of the generation models) are introduced in the model as *additive linear regressors* over the objective variable

3.2.5.1   Demand models

Demand models consist of models that correlate the demand readouts with the typology of the day they take place. Eight different types of days are considered by SMAC: Monday to Sunday, plus an additional "holiday" day type. The input data required for the demand forecast model consists of:

- Historical demand data: **energy demand** values for every unmanaged load to be modelled, with a temporal resolution similar to the resolution required by the forecast (proposed to be 15 minutes)

- Type of day: derived directly from the timestamp of the readout. *python-holidays*[5] package is used to automatically retrieve the holidays for the region the EVSEs are located

3.2.5.2 RES generation models

RES generation models consist of models that correlate weather data with the energy generation readouts. The input data required for the RES generation forecast model consists of:

- Historical generation data: **energy generation** values for every generation unit to be modelled, with a temporal resolution similar to the resolution required by the forecast (proposed to be 15 minutes)
- Historical weather data (exogenous variables): historical values of **solar irradiance** and **temperature** in the area the generation units are located, with the same indications for temporal resolution as above. SMAC internally uses the location of the EVSEs (which are known) in order to automatically retrieve the corresponding weather information

3.2.6 Optimizer

The optimizer component is the core of SMAC, and implements a linear optimization model to derive the smart charging profiles that minimize the energy-related costs while fulfilling the provided constraints. The optimization is performed in time slots of predefined duration, which establishes the granularity of the results (e.g., 15 minutes slots). Relevant details of the model are provided in section 5.3 of D4.2 [1]. The only relevant difference with respect to the design is the consideration of presence of local storage in the model.

3.2.6.1 Inputs

Category	Parameter	Description	Optional	Default value
General	slotDuration	Duration of the slots (minutes)	Mandatory	
	timeHorizon	Time horizon for the optimization (minutes)	Mandatory	
	demand	Forecasted demand per slot, not considering the EVSEs (Wh)	X	[0,...,0] (scenario with no unmanaged demand)

---

[5] https://python-holidays.readthedocs.io/en/latest/

Category	Parameter	Description	Optional	Default value
	production	Forecasted production per slot (Wh)	X	[0,…,0] (scenario with no RES production)
	supplyPointPMax	Maximum power at the supply point, i.e. considering total demand and production (W)	X	1TW (unlimited in practical terms)
	supplyPointPMin	Minimum power at the supply point, particularly relevant on scenarios with production and V2G (W)	X	-1TW (unlimited in practical terms)
	price	Energy price per slot (currency units). Applies only to energy taken from the supply point (i.e. local generation is modelled with price 0)	X	[0,…,0] (no price considerations are taken into account in the optimization)
	opportunityCost	Opportunity cost faced by each occupied EVSE and slot	X	0 (long-lasting sessions will not be discouraged)
	flexOrders	Flexibility orders to be met per slot (extra reduction of supplyPointPMax)	X	[0,…,0] (no flexibility orders to be met)
Local storage	batt_capacity	Capacity per battery (Wh)	X	No batteries included in the model
	batt_soc	Current SoC per battery (Wh)	Mandatory if local storage is included in the model	

Category	Parameter	Description	Optional	Default value
	batt_chargePower	Max. charge power per battery (W)	Mandatory if local storage is included in the model	
	batt_dischargePower	Max. discharge power per battery (W)	Mandatory if local storage is included in the model	
	batt_minSoC	Minimum SoC allowed by any battery (%)	X	20%
	batt_maxSoC	Maximum SoC allowed by any battery (%)	X	100%
	batt_currentChargingProfiles	Current setpoint applied to the batteries (W)	X	If omitted, cost of initial slot won't be considered to flatten charge/discharge curves
Charge Points	ev_capacity	Capacity per EV (Wh)	Mandatory	
	ev_soc	Current SoC per EV (Wh)	Mandatory	
	ev_chargePower	Nominal charge power per EVSE (W)	Mandatory	
	ev_dischargePower	Nominal discharge power per EVSE (W)	X	[0,...,0] (no EVSE allows V2G)
	ev_targetSlot	Slot index (0-based) when each of the EVs should be charged	Mandatory	
	ev_targetSoC	Minimum SoC to be reached by each EV by the corresponding targetSlot (Wh)	Mandatory	

Category	Parameter	Description	Optional	Default value
	ev_minSoC	Minimum SoC allowed by any EV (%)	X	20%

3.2.6.2 Outputs

Parameter	Description
status	Status of the optimization process (Optimal/Infeasible)
batt_chargingProfiles	Array with the optimum charging profiles per battery (W), only if status is Optimal
ev_chargingProfiles	Array with the optimum charging profiles per EVSE (W), only if status is Optimal
upperFlexProfile	Array with the overall estimated upper flexibility availability (W), only if status is Optimal
lowerFlexProfile	Array with the overall estimated lower flexibility availability (W), only if status is Optimal

3.2.7 Weather forecast provider

As described in section 3.2.5.2, RES generation models have a dependency on weather-related data. In order to fulfil this requirement, SMAC will automatically connect to an external weather service provider in order to retrieve current and forecast weather data. The selected service is weatherbit.io[6], since it already processes and provides solar irradiance values and forecasts out-of-the-box, which is of particular relevance for the PV generation models.

---

[6] https://www.weatherbit.io/

4. SMAC documentation

This section describes how SMAC can be used by the CPOs. Two different interfaces are hereby described: a REST API, and Graphical User Interface.

4.1 REST API

This section presents the API used by SMAC to communicate with the rest of involved actors (EV drivers/EMSPs and CPOs). The implemented API mainly corresponds to the documentation provided in D4.2, with minor changes. API design is based on OCPI 2.2, with extensions where deemed necessary. In order to facilitate readability, complete specification of the implemented interfaces is documented in the following sections.

According to the different Smart Charging Topologies documented by OCPI 2.2, SMAC follows the third type – *The CPO delegated Smart Charging to SCSP*. It is therefore important that CPOs integrating with SMAC take into account the provided recommendations on notifying EV drivers about the potential variation of charging speed as an effect of the application of smart charging profiles, as well as the obligation to respect local privacy laws in the exchange of information with the SCSP.

Figure 6 OCPI 2.2 Smart Charging Topology implemented by SMAC
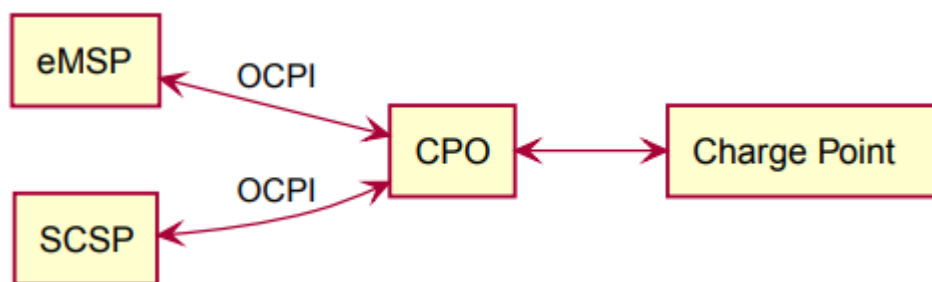
4.1.1   Interfaces implemented per actor

The following table details the interfaces to be implemented by each one of the actors in order to make use of SMAC.

Table 1 OCPI modules and roles

Module	CPO	HUB	SCSP
Sessions	Sender (extended)	Both	Receiver (extended)
ChargingProfiles	Receiver	Both	Sender
SmartChargingOptimisation (new)	Sender	Both	Receiver

Sessions module is compliant with the description of OCPI 2.2, with minor extensions. Proposed changes are documented in the following sections.

ChargingProfiles module is compliant with the description of OCPI 2.2. Reader is therefore referred to the OCPI 2.2 specification for the documentation of this module.

SmartChargingOptimization module is a new module designed as an extension to OCPI 2.2, which will enable the communication of the inputs required by SMAC in a manner that is coherent with the rest of the integration, using OCPI 2.2 guidelines.

4.1.2 Sessions module specification

The Sessions module implemented by SMAC is compliant with OCPI 2.2 specification, but extends it as deemed necessary to enable the calculation of smart charging profiles considering V2G.

4.1.2.1   ChargingPreferences object

The extension of the Sessions module consists on the inclusion of additional parameters to the ChargingPreferences object.

This object is used to communicate EV driver preferences to the SCSP. As specified in OCPI 2.2, it contains four properties, namely:

- Profile type

- Departure time (timestamp)

- Energy need (kWh)

- Discharge allowed (true/false)

An analysis of the smart charging strategies reveals that these four parameters are not enough if V2G strategies are considered. In addition to the energy needs, there is the requirement to know the current SoC of the EV battery (capacity can be derived from those two values). Even though SoC can be in theory reported by the Charging Points themselves (OCPP defines measures for SoC as part of the MeterValues message), this is not the usual case. It is therefore more reliable to get this information directly from the EV driver. With this respect, the ChargingPreferences object is extended to include the following parameters as well:

- capacity: total capacity of the EV battery (kWh)

- soc: current state of charge of the EV battery upon connection to the Charge Point (%)

- target_soc: desired SoC upon departure time (%) – actual energy needs can be derived from the combination of capacity, soc and target_soc

4.1.2.2 Remark for CPOs integrated in the INCAR Platform

CPOs integrated with the INCAR Platform do not need to send the session-related objects to SMAC. INCAR Platform will automatically redirect all necessary objects to SMAC whenever this information affects an EVSE that is part of an optimization object. CPOs will therefore automatically receive updated charging profiles from SMAC.

Figure 7 INCAR/SMAC Session object exchange
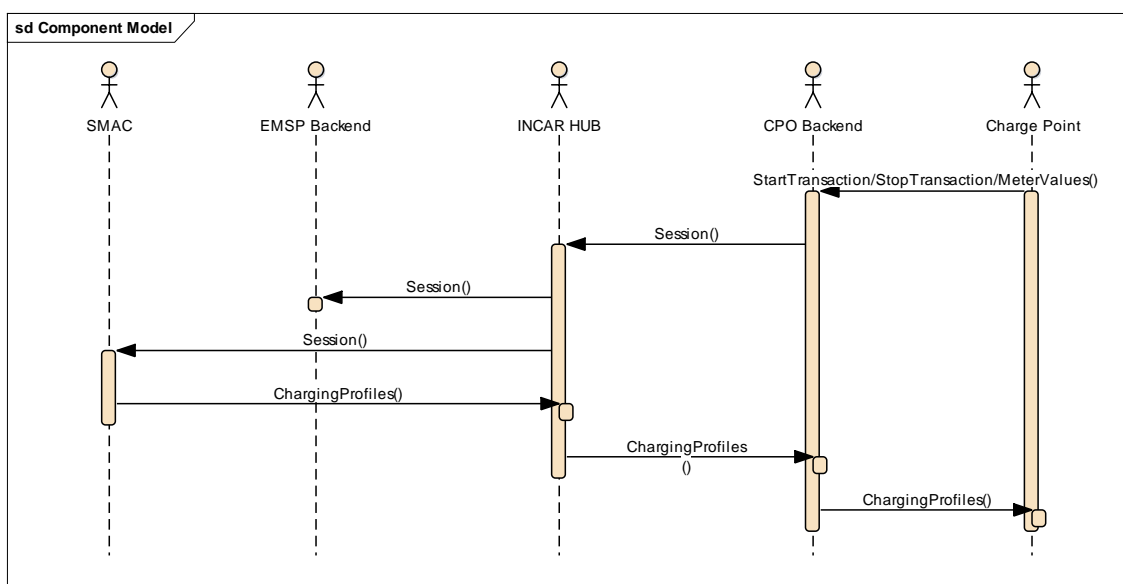
Figure 8 INCAR/SMAC ChargingPreferences object exchange
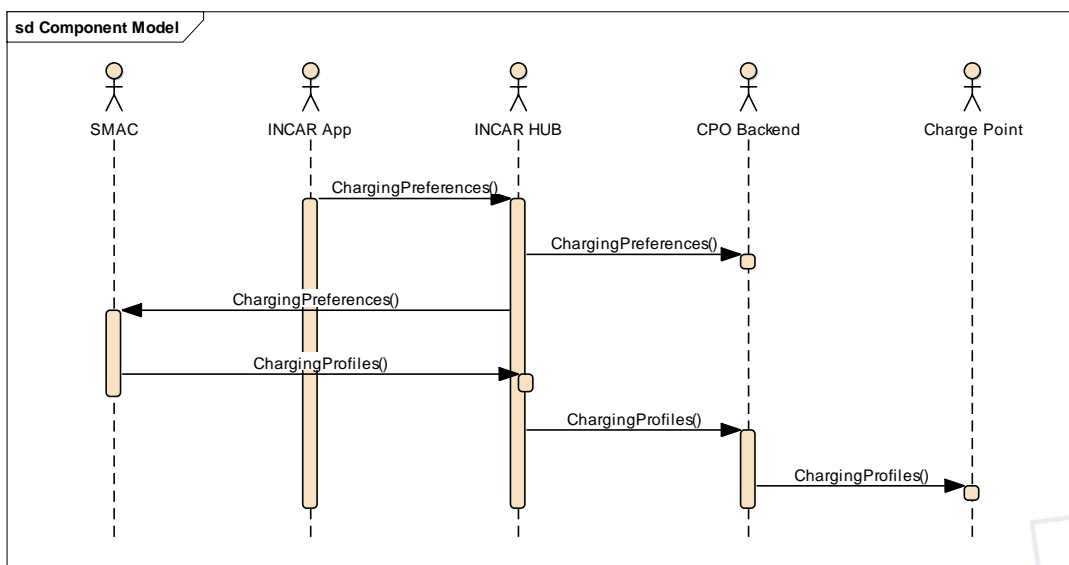
4.1.3 ChargingProfiles module specification

No changes in the ChargingProfiles module specification are proposed. Reader is therefore referred to the OCPI 2.2 specification for the documentation of this module.

4.1.4 SmartChargingOptimization module specification

The smart charging optimization module gives SCSPs knowledge of the smart charging inputs of charging stations owned by a CPO. This module implements two main functions, which are represented in the following figure:

Figure 9: Smart charging optimization module workflow
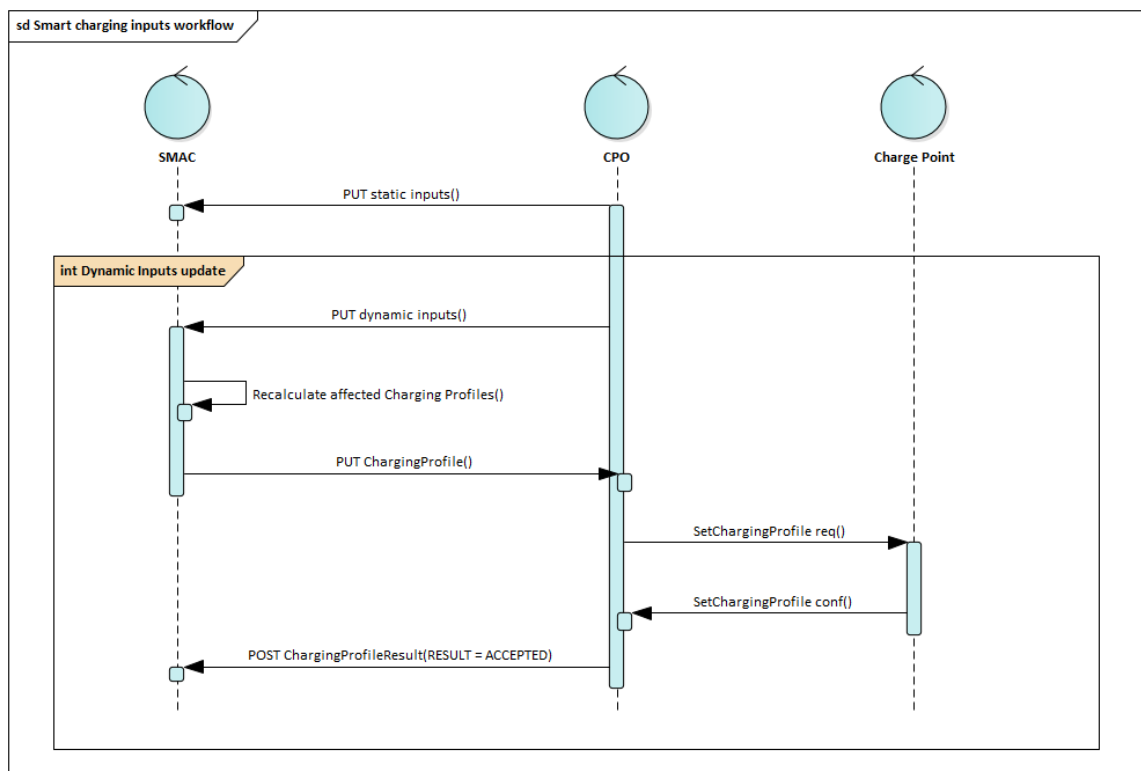
- Firstly, the CPO configures the required set of static smart charging optimization inputs. These optimization inputs define how a list of one or several EVSEs need to be considered together in the calculation of the optimum smart charging profiles (e.g., because they are connected to a common supply point with specific constraints), as well as other relevant static data to be considered by SMAC optimization algorithms. This step may take place by using the REST API, or by making use of the corresponding GUI of SMAC.

- Regularly, CPO updates dynamic information that needs to be considered by the algorithms that compute the optimum charging profiles (demand, generation, energy prices or flexibility requests)

4.1.4.1   Flow and lifecycle

4.1.4.1.1   Push model

When the CPO creates a new smart charging optimization object for a set of rechargers, the CPO sends them to the SCSPs by calling the PUT method on the SCSP smart charging optimization endpoint with the newly created object.

Any changes to the smart charging optimization object in the CPO′s system can be sent to the SCSP system by calling the PUT or the PATCH method on the SCSP smart charging optimization endpoint with the updated object.

When the CPO deletes a smart charging optimisation, CPO will update the SCSP system by calling DELETE on the SCSP smart charging optimisation endpoint with the ID of the deleted object.

When the CPO is not sure about the state or existence of a smart charging optimisation object in the system of a SCSP, the CPO can use a GET request to validate the smart charging optimisation object in the SCSP's system.

4.1.4.1.2   Pull model

SCSPs who do not support the Push model need to call GET on the CPO′s smart charging optimisation endpoint to receive all smart charging optimisation data.

4.1.4.2   Interfaces and endpoints

There is both a Sender and a Receiver interface for smart charging optimisation. Advised is to use the push direction from Sender to Receiver during normal operation.

4.1.4.2.1   Sender interface

Typically implemented by market roles like CPOs.

The Sender's smart charging optimisation interface gives the Receiver the ability to request smart charging optimisation information.

Table 2: Smart charging optimisation module sender interface

Method	Description
GET	Returns smart charging optimisation objects, last updated between the {date_from} and {date_to} (paginated)
POST	n/a
PUT	n/a
PATCH	n/a

Method	Description
DELETE	n/a

4.1.4.2.1.1 GET method

Fetch information about all smart charging optimisations.

Endpoint structure definition:

```
{smartChargingOptimisation_endpoint_url}?[date_from={date_from}]&[date_to={date_to}]&[offset={offset}]&[limit={limit}]
```

Examples:

```
https://www.server.com/ocpi/cpo/2.2/smartChargingOptimisation/?date_from=2019-01-8T12:00:00&date_to=2019-01-29T12:00:00
```

```
https://ocpi.server.com/2.2/smartChargingOptimisation/?offset=50
```

```
https://www.server.com/ocpi/2.2/smartChargingOptimisation/?date_from=2019-01-29T12:00:00&limit=100
```

```
https://www.server.com/ocpi/cpo/2.2/smartChargingOptimisation/?offset=50&limit=100
```

Request Parameters

If additional parameters: {date_from} and/or {date_to} are provided, only smart charging optimisations with last_updated between the given {date_from} (including) and {date_to} (excluding) will be returned.

This request is paginated; it supports the pagination related URL parameters.

Table 3: Smart charging optimisation – Sender GET request parameters

Parameter	Datatype	Required	Description
date_from	DateTime	no	Only return smart charging optimisations that have last_updated after or equal to this Date/Time (inclusive).
date_to	DateTime	no	Only return smart charging optimisations that have last_updated up to this Date/Time, but not including (exclusive).
offset	int	no	The offset of the first object returned. Default is 0.
limit	int	no	Maximum number of objects to GET.

Response Data

The endpoint returns an object with a list of valid smart charging optimisations; the header will contain the pagination related headers.

Any older information that is not specified in the response is considered no longer valid. Each object must contain all required fields. Fields that are not specified may be considered as null values.

Table 4: Smart charging optimisation – Sender GET response data

Type	Cardinality	Description
SCOptimisation	*	List of all smart charging optimisations

4.1.4.2.2 Receiver interface

Typically implemented by market roles like: SCSP.

Smart charging optimisations are Client Owned Objects, so the endpoints need to contain the required extra fields: {party_id} and {country_code}.

Endpoint structure definition:

{smartChargingOptimisation_endpoint_url}/{country_code}/{party_id}/{optimisation_id}

Example:

```
https://www.server.com/ocpi/scsp/2.2/smartChargingOptimisation/BE/BEC/12
```

Table 5: Smart charging optimisation module receiver interface

Method	Description
GET	Retrieve a smart charging optimisation as it is stored in the SCSP's system
POST	n/a
PUT	Push new/updated smart charging optimisation object to the SCSP
PATCH	Update existing smart charging optimisation object
DELETE	Remove a smart charging optimisation object which is no longer in use and will not be used in future either

4.1.4.2.2.1 GET method

This method allows any CPO to retrieve back the static information about any smart charging optimisation scenario already registered in SMAC.

Request parameters

The following parameters can be provided as URL segments

Table 6: Smart charging optimisation – Receiver GET request parameters

Parameter	Datatype	Required	Description
country_code	CiString(2)	yes	Country code of the CPO performing the GET request on the SCSP's system.
party_id	CiString(3)	yes	Party ID (Provider ID) of the CPO performing the GET request on the SCSP's system.
id	CiString(36)	yes	Optimisation.id of the smart charging optimisation object to retrieve.

Response data

The response contains the requested object.

Table 7: Smart charging optimisation – Receiver GET response data

Type	Cardinality	Description
SCOptimisation	1	The requested smart charging optimisation object

4.1.4.2.2.2   PUT method

New or updated smart charging optimisation objects are pushed from the CPO to the SCSP.

Request Body

In the PUT request, the new or updated smart charging optimisation object is sent in the body.

Table 8: Smart charging optimisation – Receiver PUT body

Type	Cardinality	Description
SCOptimisation	1	The requested smart charging optimisation object

Request parameters

The following parameters can be provided as URL segments.

Table 9: Smart charging optimisation – Receiver PUT URL segments

Parameter	Datatype	Required	Description
country_code	CiString(2)	yes	Country code of the CPO performing the PUT request on the SCSP's system.
party_id	CiString(3)	yes	Party ID (Provider ID) of the CPO performing the PUT request on the SCSP's system.

Parameter	Datatype	Required	Description
id	CiString(36)	yes	Optimisation.id of the smart charging optimisation object to create or replace.

Example: add a smart charging optimisation object

```
PUT To URL: https://www.server.com/ocpi/scsp/2.2/optimisation/BE/BEC/1

{
      "country_code": "BE",
      "party_id": "BEC",
      "id": "1",
      "evses": [{
            "location_id": "1",
            "evse_uid": "1",
            "connectors": [{
                  "connector_id": "1",
                  "power": 5000,
                  "discharge_power": 0,
            },
            {
                  "connector_id": "2",
                  "power": 5000,
                  "discharge_power": 100,
      }],

      "max_power": 10000,
      "min_power": 0,
      "opportunity_cost": 10,
      "last_updated": "2015-06-29T22:39:09Z"
}
```

4.1.4.2.2.3   PATCH method

Same as the PUT method, but only the fields/objects that need to be updated have to be present. Fields/objects which are not specified are considered unchanged.

Any request to the PATCH method SHALL contain the last_updated field.

Example: update optimisation inputs

```
PATCH https://www.server.com/ocpi/scsp/2.2/optimisation/NL/TNM/1

{
"demand":[{
      "value": 4000,
      "time_slot": "2015-06-29T22:49:09Z"
},{
      "value": 5000,
```

```
        "time_slot": "2015-06-29T23:49:09Z"
}],
"generation":[{
        "value": 3000,
        "time_slot": "2015-06-29T22:49:09Z"
},{
        "value": 3500,
        "time_slot": "2015-06-29T23:49:09Z"
}],
"price":[{
        "value": 0.1,
        "time_slot": "2015-06-29T22:49:09Z"
},{
        "value": 0.5,
        "time_slot": "2015-06-29T23:49:09Z"
}],
"flex_orders":[{
        "value": 5000,
        "time_slot": "2015-06-29T22:49:09Z"
},{
        "value": 5000,
        "time_slot": "2015-06-29T23:49:09Z"
}],
"last_updated": "2015-06-29T22:39:09Z"
}
```

4.1.4.2.2.4   DELETE method

Delete a smart charging optimisation object that is not used anymore and will not be used in future either.

Request parameters

The following parameters can be provided as URL segments.

Table 10: Smart charging optimisation – Receiver DELETE request parameters

Parameter	Datatype	Required	Description
country_code	CiString(2)	yes	Country code of the CPO performing the DELETE request on the SCSP's system.
party_id	CiString(3)	yes	Party ID (Provider ID) of the CPO performing the DELETE request on the SCSP's system.
id	CiString(36)	yes	Optimisation.id of the smart charging optimisation object to delete.

4.1.4.3   Object description

4.1.4.3.1   SCOptimisation object

The SCOptimisation object contains all the details required to properly define an optimisation context in SMAC. This includes information about the CPO identification, details of the supply point (maximum capacity, maximum allowed power injection), identification of the EVSEs that take part of the optimization and other optional elements to be taken into account, such as the existence of local storage asset and its capacity. Additionally, the object can be used to notify dynamic data (i.e., data related to demand, generation, prices or flexibility orders) to SMAC.

Table 11: Smart charging optimisation object

Property	Type	Cardinality	Description
country_code	CiString(2)	1	ISO-3166 alpha-2 country code of the CPO that 'owns' this optimisation context.
party_id	CiString(3)	1	CPO ID of the CPO that 'owns' this optimisation context (following the ISO-15118 standard).
id	CiString(36)	1	The unique id that identifies the optimisation context in the CPO platform.
storage_capacity	number	?	Capacity of stationary local storage, if present (Wh)
max_power	number	?	Maximum power to be delivered simultaneously, considering non-controllable demand, production and EVSE-related demand (W)
min_power	number	?	Minimum power at the supply point, particularly relevant on scenarios V2G (W)
evses	Evse	+	EVSEs to apply the smart charging optimisation
opportunity_cost	number	?	Linearly-increasing additional cost considered to disincentive long-lasting charging sessions [€/slot]
demand	Demand	*	Local demand (one element per optimisation time slot)
generation	Generation	*	Local generation (one element per optimisation time slot)
demand_forecast	Demand	*	Local demand forecasts (one element per optimisation time slot)
generation_forecast	Generation	*	Local generation forecasts (one element per optimisation time slot)
storage_soc	SOC	*	Local storage state of charge (one element per optimisation time slot)
price	Price	*	Energy price (one element per optimisation time slot)

Property	Type	Cardinality	Description
flex_orders	FlexOrders	*	Grid operator flexibility orders to be fulfilled
last_updated	DateTime	1	Timestamp when this object was last updated (or created)

4.1.4.3.1.1 Examples

SCOptimisation example

```
{
      "country_code": "BE",
      "party_id": "BEC",
      "id": "1",
      "evses": [{
           "location_id": "1",
           "evse_uid": "1",
           "connectors": [{
                "connector_id": "1",
                "power": 5000,
                "discharge_power": 0,
           },
           {
                "connector_id": "2",
                "power": 5000,
                "discharge_power": 100,
      }],
      "demand":[{
           "value": 4000,
           "time_slot": "2015-06-29T22:49:09Z"
      },{
           "value": 5000,
           "time_slot": "2015-06-29T23:49:09Z"
      }],
      "generation":[{
           "value": 3000,
           "time_slot": "2015-06-29T22:49:09Z"
      },{
           "value": 3500,
           "time_slot": "2015-06-29T23:49:09Z"
      }],
      "price":[{
           "value": 10,
           "time_slot": "2015-06-29T22:49:09Z"
      },
      {
           "value": 5,
           "time_slot": "2015-06-29T23:49:09Z"
      }],
      "flex_orders":[{
           "value": 5,
           "time_slot": "2015-06-29T22:49:09Z"
```

```
      },
      {
            "value": 5,
            "time_slot": "2015-06-29T23:49:09Z"
      }],
      "max_power": 10000,
      "min_power": 0,
      "opportunity_cost": 10,
      "last_updated": "2015-06-29T22:39:09Z"
}
```

4.1.4.4   Data types

4.1.4.4.1   EVSE

Table 12: EVSE object

Property	Type	Cardinality	Description
location_id	CiString(36)	1	Location.id of the Location (belonging to the CPO) on which the optimisation context should be applied.
evse_uid	CiString(36)	1	EVSE.uid of the charge point (belonging to the CPO) on which the optimisation context should be applied.
connectors	Connector	+	Set of connectors of the EVSE, including nominal charge and discharge power

4.1.4.4.2   Connector

Table 13: Connector object

Property	Type	Cardinality	Description
connector_id	number	1	Id of the connector (belonging to the CPO) on which the optimisation context should be applied.
power	number	1	Nominal power [kW]
discharge_power	number	?	Nominal discharge power [kW]

4.1.4.4.3   Demand

Table 14: Demand object

Property	Type	Cardinality	Description
time_slot	DateTime	1	Time slot where demand value applies
value	number	1	Forecasted demand [kWh]

4.1.4.4.4 Generation

Table 15: Generation object

Property	Type	Cardinality	Description
time_slot	DateTime	1	Time slot where demand value applies
value	number	1	Forecasted generation [kWh]

4.1.4.4.5 Price

Table 16: Price object

Property	Type	Cardinality	Description
time_slot	DateTime	1	Time slot where demand value applies
value	number	1	Energy price [€/kWh]

4.1.4.4.6 SOC

Table 17: SOC object

Property	Type	Cardinality	Description
time_slot	DateTime	1	Time slot where SoC has been measured
value	number	1	State of Charge [%]

4.1.4.4.7 FlexOrders

Table 18: Flexibility orders object

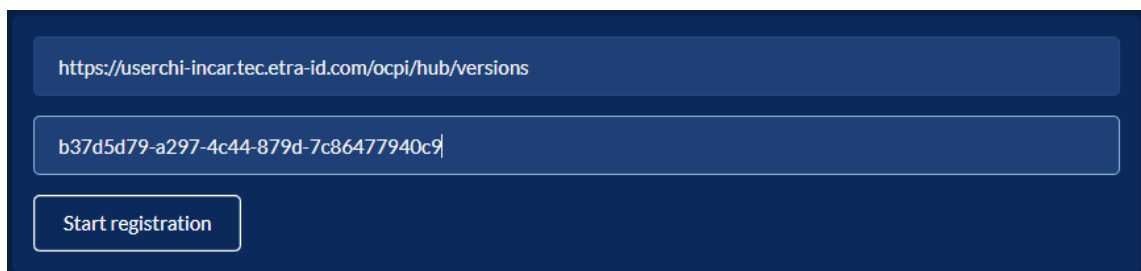| Property	Type	Cardinality	Description
time_slot	DateTime	1	Time slot where demand value applies
value	number	1	Required offsets over the maximum power at supply point (usually the contracted power) [kW]

4.2 User Interface

4.2.1 Registration in the INCAR Platform

In terms of OCPI 2.2 protocol, SMAC implements the Smart Charging Service Provider role (SCSP). It therefore includes a form that can be used to initiate the OCPI registration with the INCAR Hub, as part of the commissioning of the service.

This page displays a simple form with the URL of the Hub (or any other OCPI partner) and the access token (Token A, see OCPI registration process). Once the form is submitted, SMAC backend will initiate the OCPI registration process.

Figure 10 OCPI registration dialog

4.2.2 Configuration of Optimization Contexts

This section can be used by CPOs to configure the Optimization Contexts appropriately. In order to facilitate usability, it is integrated in the user interface of the INCAR Platform and accessible only if the user has a CPO role.

In order to create a new optimization context, the CPO will need to set up the following static inputs:

- Selection of EVSEs located under the same supply point. For each EVSE, the maximum discharge power has to be provided (if EVSE allows V2G)

- Supply point contracted capacity (W): indicates the maximum power that can be delivered at the supply point, therefore imposing a limit on the aggregation of the power consumed by the EVSEs and the local unmanageable demand (if it exists)

- Supply point injection limit (W): indicates the maximum power that can be injected back to the grid. Particularly relevant in those contexts with presence of RES generation, or allowing V2G schemes

- Local storage capacity (Wh): capacity of the local ESS (if any)

- Opportunity cost (€/slot): synthetic cost indicating the opportunity cost faced by each occupied EVSE and slot. Used by SMAC optimizer to discourage long-lasting charging sessions. Particularly useful if EVSEs are public.

Existence of unmanageable demand and/or RES generation will be automatically derived from the existence (or not) of associated measurements, communicated to SMAC via its REST API.

Other dynamic parameters (energy price, flexibility orders and local storage SoC) need to be communicated on a regular basis using the REST API.

Figure 11 New optimization context dialog

Additionally, the INCAR Platform UI lists the existing Optimization Contexts, giving access to its details and the possibility to edit them if necessary.

Figure 12 Optimization Contexts list

Figure 13 Optimization contexts details

4.2.3 Transaction details

The *Transactions* section provides access to the information of all transactions performed by any EV driver in any of the Charge Points managed by the CPO. This section, documented under the INCAR Platform deliverable [5], has been extended with additional information provided by SMAC.

Those users with a CPO role will find additional information under the *Details* view of any transaction involving a Charging Point under a SMAC optimization context. SMAC will calculate and periodically re-evaluate the most appropriate charging profile for the running session during its duration. CPO will have access to the applicable charging profile (as calculated by SMAC) for any finalised or ongoing session.

Figure 14 Transaction details
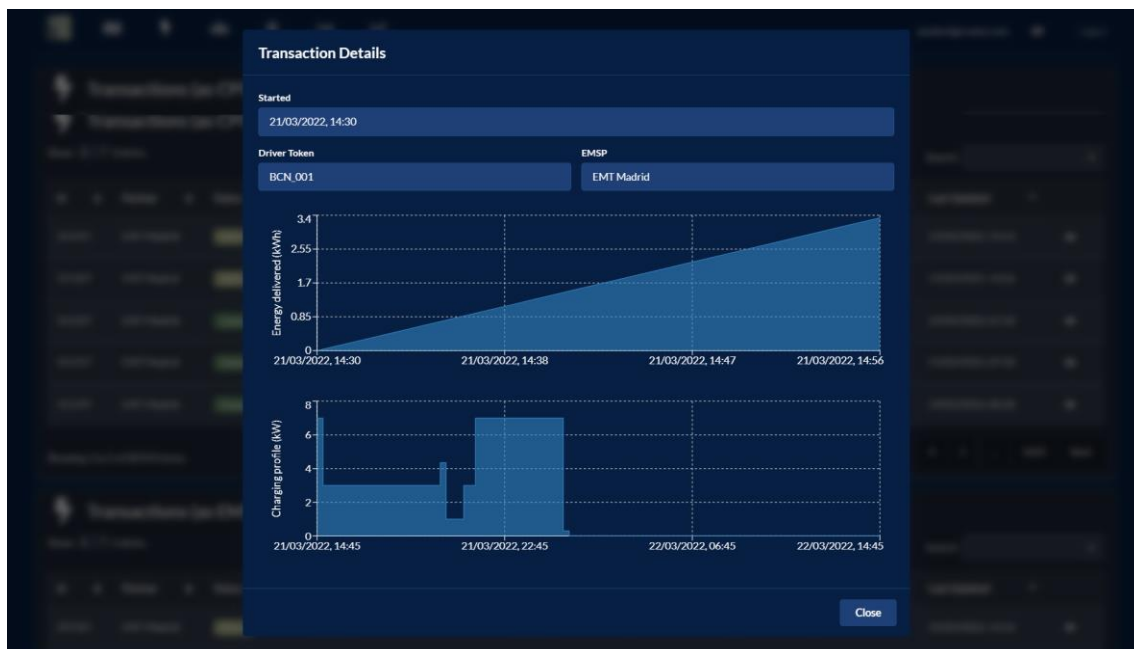
4.2.4 SMAC optimization details

Any user accessing the INCAR Platform UI with a role of CPO will have access to a new section that provides access to all the details managed by SMAC.

This section displays two selectors:

- A dropdown to select any of the *Optimization Contexts* defined by the user (CPO)

- A calendar to select a date range

Upon selection of both, all relevant information will be queried from SMAC databases and represented appropriately in four different charts. This information includes:

- Energy-related dynamic inputs, displaying registered demand, generation and their corresponding forecasts (if provided by the CPO)

- Resulting charging profiles, i.e. maximum active power to be delivered per EVSE over time

- Energy price curve (for those CPOs providing energy tariff information to SMAC)

- Flexibility orders that have been registered, i.e. temporary reduction of the maximum active power to be scheduled at the supply point, as set in the optimization context definition

Figure 15 SMAC optimization details

5. Lab-testing

5.1 Methodology

The lab-testing approach that has been followed for this tool is based on the work taken in former H2020 projects such as NOBEL GRID [6] or WiseGRID [7], as well as in the lab-testing phase of the INCAR Platform, part of the work performed in WP3 [5]. This method has shown to be successful for these types of projects, so it has been thoroughly studied and applied to take into account the specifics of the USER-CHI project.

5.1.1 Test plan

For this activity, SMAC has been deployed and connected to the staging environment of the INCAR Platform. The testing activities are all based on the following test plan:

1.   Review the project requirements and use cases

2.   Define the features to be tested from those and classify them into test groups

3.   Detail test cases for validation of named features

4.   Execute the test cases

5.   Document the test protocols

In this context, results detailed in deliverable D1.1 "User requirements for USER-CHI solutions" [8] are taken as a core input, considering that requirements and use cases are already defined and mapped to specific applications developed within the project. The focus of the tests reported in this deliverable is to put on functional testing, in preparation of the applications to be demonstrated in the upcoming piloting phase.

5.1.2 Test cases specification

All test cases specified in the following section use a template sheet as shown in the following table.

Table 19 Test case template

Name	The test case code and name that is unique to the project.		
Module under test	The devices or systems under test	Resp.	Main partner responsible for the test
Module requirement	The requirement, use case, or certification rule which is validated by the test case		
Test environment	List of elements needed for the test execution		
Features to be tested	List of features to be tested		
Features not to be tested	(Optional) Remark on features that are specifically left out of this testing procedure		
Preparation	Short list of steps needed for preparing the test environment for test execution		
Dependencies	(Optional) List of test case codes defining test cases which need to be passed before the test case at hand can be started		
Steps	Testing procedures		
Pass criteria	Expected (measurable) results, allowing to unambiguously judge if the test is passed or not passed (i.e. the product requirement was validated or not validated)		
Suspension criteria	(Optional) Conditions under which continuation of the test is considered pointless because testing results would be invalid		
Results	Short list of results		

5.1.3 Testing environment

In order to conduct the tests of the different software modules, a (private) staging installation of SMAC has been performed. Under this staging environment, SMAC has been registered as Smart Charging Service Provider on the INCAR Platform staging environment. Therefore, lab-testing takes advantage of the already existing couple of CPO and EMSPs backend systems that had been previously configured and integrated in that environment.

Additionally, specific charge point simulators have been used to simulate all the required parameters of the charge points (e.g. transaction start and end, or notification of meter values and EV driver preferences) and trigger the pre-defined test cases.
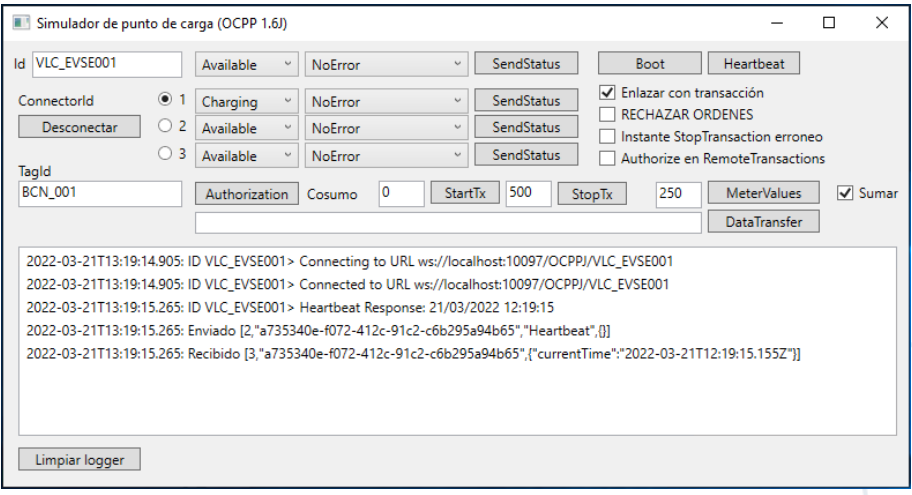
5.2 Results

This section presents the results of the tests that have been performed to validate the correct integration and operation of SMAC.

Name	SMAC registration as SCSP		
Module under test	API	Resp.	ETRA
Module requirement	SMC_004, SMC_014, SMC_019		
Test environment	Staging environment of INCAR Platform		
Features to be tested	Correct registration as INCAR Hub partner		
Features not to be tested			
Preparation	INCAR Platform up and running		
Dependencies			
Steps	1. Under INCAR Platform UI, with administrator grants, create a new TOKEN A for SMAC 2. Under SMAC UI, with administrator grants, introduce provided TOKEN A and INCAR Hub *versions* endpoint		
Pass criteria	Registration process finishes successfully according to OCPI 2.2 specification. TOKEN B and TOKEN C are created, exchanged and stored in the corresponding databases.		
Suspension criteria	Errors occur as part of the registration process		

Results	Test successful. Registration process takes place as expected. Hub connection details get successfully stored in SMAC database:
	
	{
	    "_id" : "HUBESHUB",
	    "_expired" : false,
	    "_updated" : ISODate("2022-03-03T12:41:14.802+01:00"),
	    "business_details" : {
	        "name" : "USERCHI HUB"
	    },
	    "country_code" : "ES",		

```
        "endpoints" : [

                {

                        "identifier" : "credentials",

                        "role" : "RECEIVER",

                        "url"        :        "https://userchi-
incar.tec.etra-id.com/ocpi/2.2/credentials/receiver"

                },
[…]

                {

                        "identifier" : "scoptimizations",

                        "role" : "RECEIVER",

                        "url"        :        "https://userchi-
incar.tec.etra-id.com/ocpi/2.2/scoptimization"

                }

        ],

        "last_updated"        :        ISODate("2022-03-
03T12:41:14.795+01:00"),

        "party_id" : "HUB",

        "role" : "HUB",

        "status" : "CONNECTED",

        "token" : "addcaa5b-eb65-47ed-800f-xxxxxxxxxxxx",

        "tokenSmac"     :       "17efd15e-8cb2-499f-ac05-
xxxxxxxxxxxx"

}
```

Name	Session updates are successfully communicated to SMAC		
Module under test	API Optimization contexts database	Resp.	ETRA
Module requirement	SMC_002		
Test environment	Staging environment of INCAR Platform		

	Charging Point simulator
Features to be tested	Changes initiated by the Charging Point produce data updates in SMAC
Features not to be tested	
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Charging Point simulator up and running Optimization context is configured, and includes the simulated Charging Point
Dependencies	SMAC registration as SCSP
Steps	1. Using Charging Point simulator, trigger a Start Transaction message 2. Using Charging Point simulator, trigger a Meter Values message 3. Using Charging Point simulator, trigger a Stop Transaction message
Pass criteria	Actions initiated at Charging Point level get communicated to the INCAR Hub as Session updates Session updates referring to Charging Points managed by SMAC reach SMAC
Suspension criteria	Session updates do not reach SMAC
Results	Test successful Screenshot of the Charging Point simulator

	All expected updates reach SMAC appropriately (tested using breakpoints during a debugging session)

Name	Driver preferences reach SMAC		
Module under test	API Optimization contexts database	Resp.	ETRA
Module requirement	SMC_002		
Test environment	Staging environment of INCAR Platform POSTMAN		
Features to be tested	EV driver preferences triggered by the INCAR mobile app (or EMSP) produce data updates in SMAC		
Features not to be tested			
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Charging Point simulator up and running Optimization context is configured, and includes the simulated Charging Point		
Dependencies	SMAC registration as SCSP Session updates are successfully communicated to SMAC		
Steps	1. Initiate a simulated charging session, as specified in test case *Session updates are successfully communicated to SMAC* 2. Using POSTMAN, emit a driver preferences message, similarly as EMSP or INCAR mobile app would operate		
Pass criteria	This action produces a data update in SMAC EV driver preferences get stored in the database of SMAC for further usage		
Suspension criteria	EV driver preferences updates do not reach SMAC		
Results	Test successful Screenshot of POSTMAN, used to mimic EMSP/INCAR app operation		

```
PUT  ⌄   http://linuxdev.lab.id:8744/ocpi/hub/2.2/sessions/ES/VLC/30803T/charging_p       Send  ⌄

Params   Auth   Headers (11)   Body ●   Pre-req.   Tests   Settings                        Cookies

raw  ⌄     JSON  ⌄                                                                          Beautify

1  {
2      "profile_type": "REGULAR",
3      "departure_time":"2022-03-05T12:00:00Z",
4      "energy_need":30,
5      "discharge_allowed": false
6  }
```

All expected updates reach SMAC appropriately (tested using breakpoints during a debugging session). Details get stored in databases, attached to the corresponding session information.

```
{

        "_id" : "ES%23VLC%2330804T",

        "_expired" : false,

        "_updated"              :              ISODate("2022-03-
04T10:42:40.485+01:00"),

        "auth_method" : "WHITELIST",

        "cdr_token" : {

                "uid" : "BCN_003",

                "type" : "RFID",

                "contract_id" : "3"

        },
[…]

        "charging_preferences" : {

                "profile_type" : "REGULAR",

                "departure_time"     :     ISODate("2022-03-
04T12:00:00.000+01:00"),

                "energy_need" : 7,

                "discharge_allowed" : false

        }

}
```

Name	Charging profiles updates are calculated upon input updates

Module under test	Orchestrator Optimizer Time series database	Resp.	ETRA
Module requirement	SMC_002, SMC_007, SMC_008, SMC_020		
Test environment	Staging environment of INCAR Platform		
Features to be tested	Updates initiated in previous test cases trigger SMAC recalculation of charging profiles		
Features not to be tested			
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Charging Point simulator up and running Optimization context is configured, and includes the simulated Charging Point		
Dependencies	Session updates are successfully communicated to SMAC Driver preferences reach SMAC		
Steps	1. Execute test case *Session updates are successfully communicated to SMAC* 2. Execute test case *Driver preferences reach SMAC*		
Pass criteria	SMAC recalculates charging profiles of affected Charging Points		
Suspension criteria	EV driver preferences updates do not reach SMAC		
Results	Test successful Screenshot of time series database, where updated charging profiles are stored 		

Name	Charging profiles are communicated back to corresponding CPOs		
Module under test	Orchestrator API	**Resp.**	ETRA
Module requirement	SMC_002		
Test environment	Staging environment of INCAR Platform		
Features to be tested	Recalculated charging profiles are sent back to the corresponding CPO backend, using the corresponding OCPI 2.2 messages		
Features not to be tested	Actual implementation of the charging profiles (i.e. communication of the profiles to the Charging Points) is responsibility of the corresponding CPO backends, out of scope of SMAC.		
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Charging Point simulator up and running Optimization context is configured, and includes the simulated Charging Point		
Dependencies	Charging profiles updates are calculated upon input updates		
Steps	1. Execute test case *Charging profiles updates are calculated upon input updates*		
Pass criteria	SMAC automatically sends the updates to the chargingprofiles module of the INCAR platform INCAR platform redirects message to the appropriate CPO backend		
Suspension criteria	Charging profiles do not reach CPO backend		

Results	Test successful
	
	INCAR platform stores the emitted messages in its database for traceability reasons. An example is provided hereby.
	
	`{`
	
	`    "_id" : "ESVLC31101T",`
	
	`    "_expired" : false,`
	
	`    "_updated"          :          ISODate("2022-03-18T12:02:32.107+01:00"),`
	
	`    "charging_profile" : {`		

```
            "start_date_time"    :    ISODate("2022-03-
18T12:02:29.557+01:00"),

            "duration" : 86400,

            "charging_rate_unit" : "W",

            "charging_profile_period" : [

                {

                        "start_period" : 0,

                        "limit" : 3060

                },
[…]

                {

                        "start_period" : 85500,

                        "limit" : 0

                }

            ]

        },

        "country_code" : "ES",

        "party_id" : "VLC",

        "response_url"  :  "https://userchi-smac.tec.etra-
id.com/ocpi/scsp/chargingprofiles/PUT/ESVLC31101T"

}
```
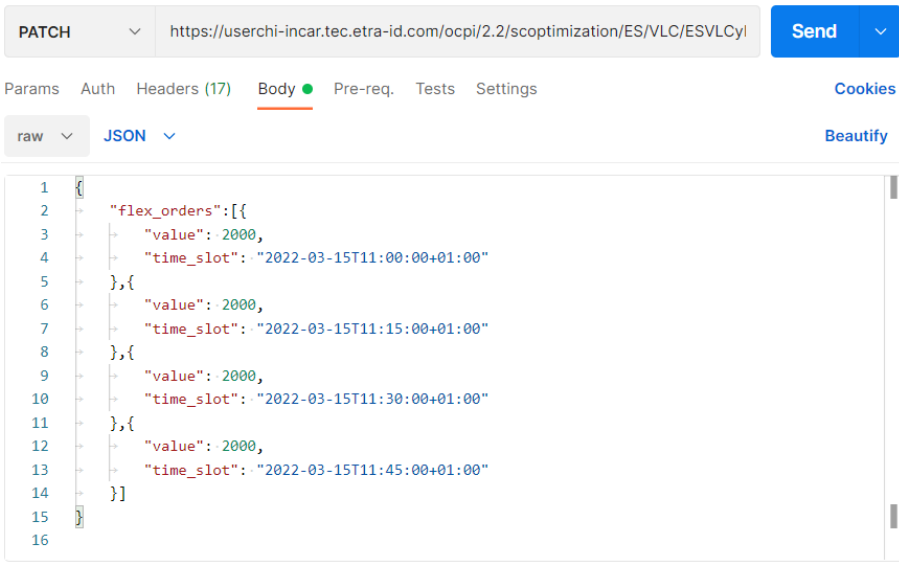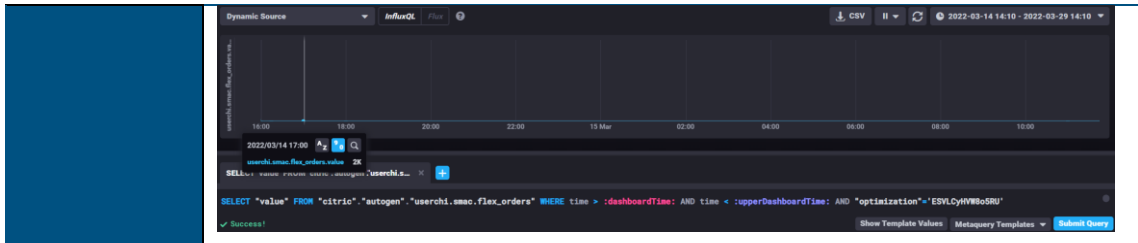
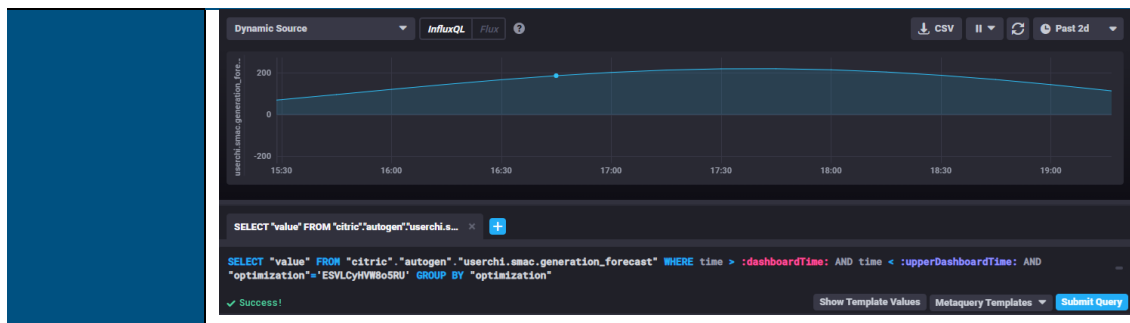| Name	Charging profiles get registered in database, linked to EVSE and Session (and CPO)		
Module under test	Time series database	Resp.	ETRA
Module requirement	SMC_021		
Test environment	Staging environment of INCAR Platform		
Features to be tested	All calculated charging profiles gets stored in the time series database, linked to the EVSE and Session Id, to facilitate further exploitation of the information		

Features not to be tested	
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Charging Point simulator up and running Optimization context is configured, and includes the simulated Charging Point
Dependencies	Charging profiles updates are calculated upon input updates
Steps	1. Execute test case *Charging profiles updates are calculated upon input updates* 2. Access time series database inspector 3. Filter by EVSE id 4. Filter by Session id
Pass criteria	Results are accessible filtering by EVSE id and Session id
Suspension criteria	
Results	Test successful Screenshots of queries and corresponding results represented hereby.

Name	Dynamic inputs reach SMAC		
Module under test	API Time series database	Resp.	ETRA

|	
Module requirement	SMC_002, SMC_009, SMC_020
Test environment	Staging environment of INCAR Platform POSTMAN
Features to be tested	All dynamic inputs sent with the OCPI 2.2 proposed extension reach SMAC and get stored in the time series database for further usage
Features not to be tested	
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Optimization context is configured
Dependencies	SMAC registration as SCSP
Steps	1. Use POSTMAN to emit updates including demand, generation, prices and flexibility orders
Pass criteria	Data reaches SMAC and gets successfully stored in the time series
Suspension criteria	Data updates do not reach SMAC
Results	Test successful Screenshot of POSTMAN, with an example of a message including flexibility order data Screenshot of time series database inspector, showing stored data accordingly

| Name	Forecast models are automatically built based on dynamic inputs		
Module under test	Weather forecast provider Forecaster Orchestrator	Resp.	ETRA
Module requirement	SMC_002		
Test environment	Staging environment of INCAR Platform POSTMAN		
Features to be tested	Whenever *demand* or *generation* dynamic inputs are sent to a SMAC optimization context for the first time, a specific forecast model gets automatically set up		
Features not to be tested			
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Optimization context is configured		
Dependencies	Dynamic inputs reach SMAC		
Steps	Use POSTMAN to emit updates including demand and generation		
Pass criteria	New forecast models get registered in SMAC database Forecast model periodically produces results (forecasted demand and generation), which get stored in the time series database for further usage		
Suspension criteria			
Results	Test successful SMAC behaves as expected, registering new forecast models on-the-fly as required. Forecasts get periodically triggered, and results appear in the database		

Name	Smart charging takes dynamic inputs into account		
Module under test	Optimizer	Resp.	ETRA
Module requirement	SMC_002, SMC_007, SMC_009, SMC_020		
Test environment	Staging environment of INCAR Platform POSTMAN		
Features to be tested	Already registered dynamic data is taken into account as an input to the optimizer module		
Features not to be tested			
Preparation	INCAR Platform up and running CPO connected to INCAR Platform Optimization context is configured		
Dependencies	Dynamic inputs reach SMAC Forecast models are automatically built based on dynamic inputs		
Steps	1. Use POSTMAN to emit updates including demand, generation, prices and flexibility orders		
Pass criteria	Dynamic inputs get used by the optimizer module Results provided are coherent with those inputs		
Suspension criteria			
Results	Test successful The screenshot below is a representative example. A flexibility order of 2kW has been established between 20:00 and 21:00. This means that, during that interval, maximum active power to be used by the EVSEs is reduced from		

10kW to 8kW. This is represented in the stacked visualization of the calculated charging profiles.
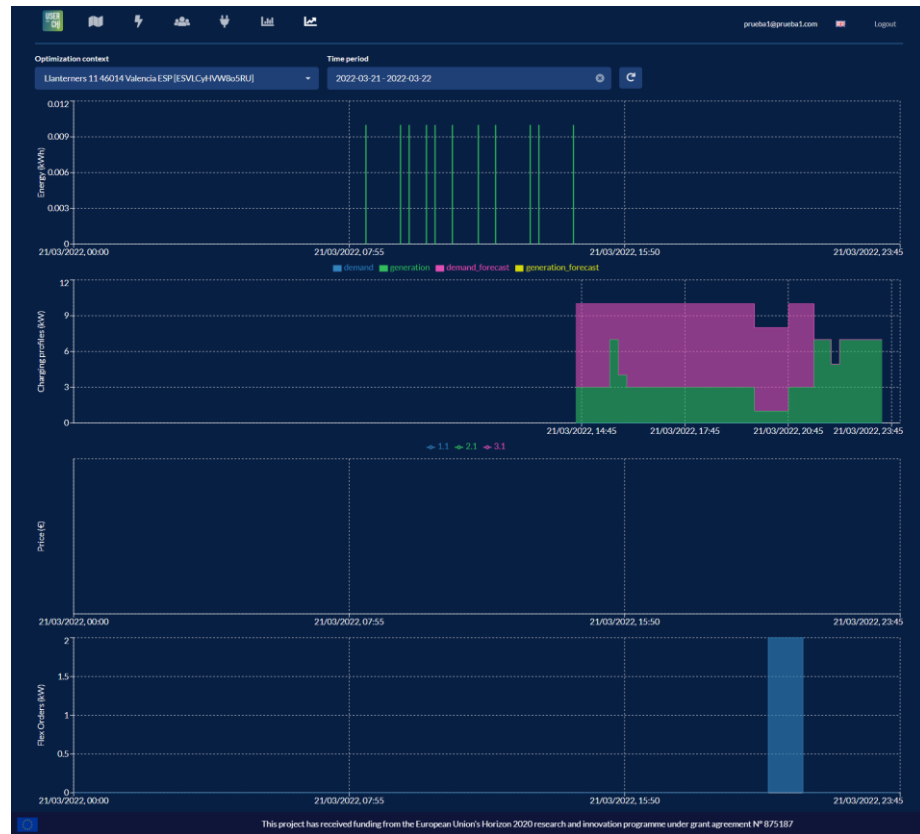
Name	All information managed by SMAC is displayed in UI		
Module under test	Graphical User Interface	Resp.	ETRA
Module requirement	SMC_021		
Test environment	Staging environment of INCAR Platform		
Features to be tested	CPOs have access to all information managed by SMAC in the appropriate sections of INCAR UI		
Features not to be tested			
Preparation	INCAR Platform up and running		
	CPO connected to INCAR Platform		
	Optimization context is configured		

Dependencies	SMAC registration as SCSP
Steps	1. Access INCAR UI with a CPO role
Pass criteria	Historical information of SMAC-related data is accessible
Suspension criteria	
Results	Test successful SMAC-related information is displayed in the following sections: - Transaction details - Optimization context details
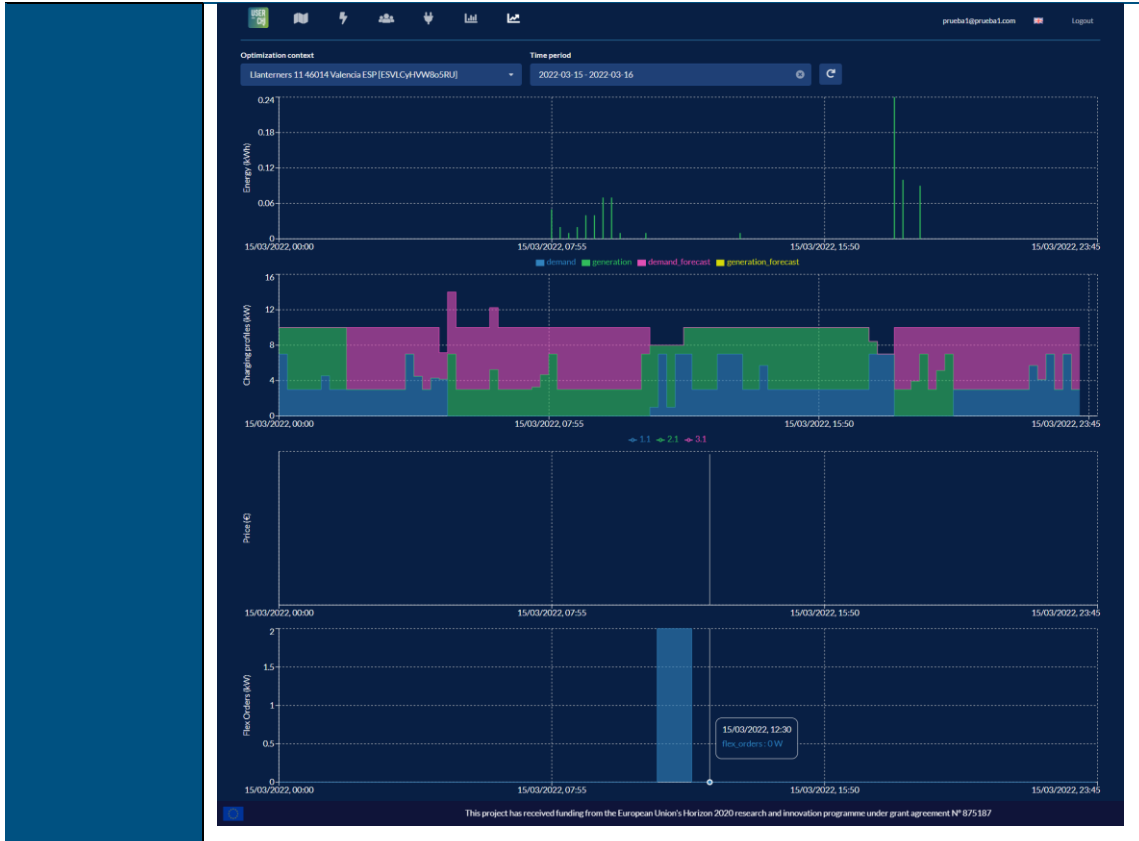
6. Pilot site integration

This section includes a short description of the status and efforts taken so far per pilot site in relation to the integration and demonstration facilitation of SMAC.

6.1 Barcelona

AMB as a local public administration is promoting electromobility in the Barcelona Metropolitan Area. AMB has deployed during the last 5 years a public charging infrastructure consisting in 10 charging stations in different municipalities of the Metropolitan Area. AMB, as a CPO and EMSP, has a service provider that operates and maintains this infrastructure thought a public contract. This provider uses ETRA backend to operate AMB's charging infrastructure.

To demonstrate USER-CHI products, INCAR and SMAC, AMB, with the product leader requirements, has planned different locations depending on the product features that want to be demonstrated.

The main location of the tests will be in Cornellà de Llobregat. In this municipality, there is one of the firsts AMB charging stations and it is very well known over the EV users and well connected. Nowadays this station has one quick charging point. To be able to demonstrate INCAR and SMAC tools, AMB enlarges this station with two new charging points. In this case, AMB is building an ultra-fast charging point (DC) and a normal charging point (AC) adjacent to the existing equipment (DC-AC). This new equipment will be an up-to-date infrastructure capable to support smart charging profiles coming from the SMAC tool. AMB has already finished the tendering process to build the infrastructure and it is nowadays working on the technical documentation, the administrative permits and the collaboration agreement with the municipality involved and is finishing technical details with the grid operator. This demo site is expected to be finished on time for the demonstration.

Within the scope of the demo in Barcelona, it has been decided to go beyond the initial scope of the demo, taking advantage of the work performed by CIRCONTROL and ETRA. It is envisioned to demonstrate SMAC in an advanced manner, by taking advantage of the features offered by OCPP 2.0 and ISO15118. The combination of those two protocols enables advanced communications between the EV and the Charge Point, which in fact allow the CPO to get all details about the battery status and energy needs automatically, without human intervention. It is planned to demonstrate in Barcelona the advantages of these technology in combination with SMAC to improve the quality of the results and the operation from the EV driver perspective.

In the framework of the project CIRCONTROL has focused on updating to OCPP2.0 protocol in the scope of its Raption 50 fast-charging product. The development of specific OCPP 2.0 modules, both in the charging point and in the CPO backend, will allow to test specific SMAC functionalities based on these new features offered by the latest version of the protocol. As CIRCONTROL is a manufacturer of charging points and provides them also to AMB, there are two locations with DC Raption 50 equipment already installed (in Barberà del Vallès and Montcada i Reixac) that can be upgraded, once the development, the laboratory and integration

tests of OCPP2.0 has been completed, in order to carry out these additional tests in the demo of Barcelona.

Figure 16 Barcelona INCAR and SMAC demo sites overview

ETRA, as provider of the backend that AMB uses to operate their Charging Points, has also already started the implementation of the OCPP 2.0 protocol on the CPO backend system. This version of the protocol consists on a major upgrade over the previous version (1.6). Focus is put in those modules necessary to implement smart charging and retrieve the inputs required by SMAC automatically from the vehicle. The CPO backend will then be in position of emitting the *ChargingPreferences* object on behalf of the EV driver, by using the data taken from the vehicle.

In more detail, a set of OCPP 2.0 Functional Blocks and Use Cases that enable the SMAC demonstration targets have been selected to be implemented by CIRCONTROL (at Charge Point side) and ETRA (at CPO backend side), whose development is still ongoing. In summary, it includes the UCs enabling the following functionalities:

- Booting a Charging Station (selection of UCs under *B. Provisioning* functional block)

- Availability and Monitoring Events (selection of UCs under *G. Availability* functional block)

- Configuring a Charging Station and Resetting a Charging Station (selection of UCs under *B. Provisioning* functional block)

- Transaction mechanism (selection of UCs under *E. Transactions* functional block)

- Notification of SoC from EV to Charging Point, and implementation of configuration variables related to MeterValues messages (selection of UCs under *J. MeterValues* functional block)

- Notification of energy requirements from EV to Central System, and execution of charging profiles notified by the Central System (selection of UCs under *K. SmartCharging* functional block)

6.2 Berlin

The physical infrastructure in Berlin's pilot site will be fully completed by the time the demonstration of the USER-CHI products starts. There will be two demonstrations sites in Berlin, in which all the newly installed AC-chargers by Qwello will deploy and demonstrate the functions of the INCAR and SMAC-tools.

The demonstration site "Prinzenstraße" is located in the heart of Berlin Kreuzberg and offers a non-discriminatory access to a newly built parking space that is to be equipped with two AC-chargers by Qwello. A high demand for charging infrastructure at this location is expected as this charging location is frequently used by private e-drivers and mobility providers. After requesting the technical requirements from the project partners VMZ, Qwello and IKEM, Gewobag has started a tendering process and received needed building permissions. The construction and technical installation have been started end of 2021 and the site Prinzenstrasse is installed and will become operative beginning of April.

The second location is located in the East of Berlin and characterized by a high demand for parking spaces due to a nearby shopping centre "City Point Center". This provides ideal conditions for a charging hub with four AC chargers by Qwello, which will be integrated into a private car park and also made accessible to the public via the INCAR app. The call for tenders is still ongoing. The expected offers will be evaluated in the next step. In this demonstration site the parking barrier is controlled from the INCAR app, which makes the pilot site integration then complete.

Figure 17 Berlin demo sites overview
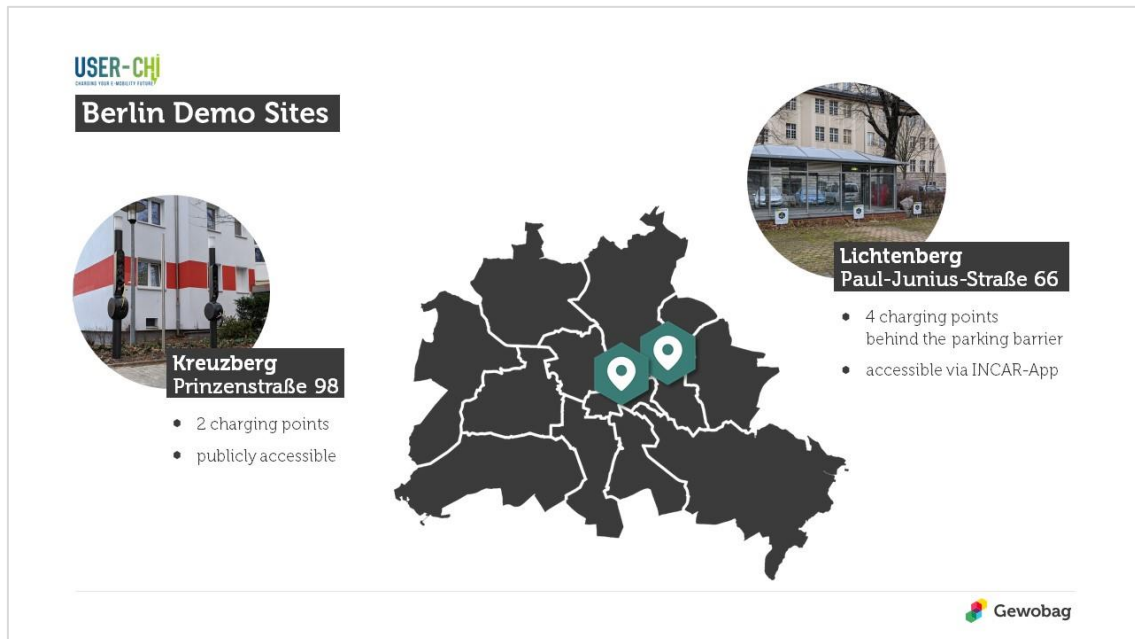
Beside the physical set-up of the demonstration sites, some integration issues evolved due to the fact that the OCPP2.2 integration is delayed due to capacity issues at Qwello with a lack of software engineers on the job market. Qwello and ETRA have agreed to install a workaround for a fast demonstration of a limited functionality of the INCAR App. In a second step Qwello would need to shift the backend to OCPI 2.2 as soon as new software engineers were able to be hired. Along the backend shift, the following modules need to be implemented:

- Hub Client Module

- SmartCharging Module

The full functionality of SMAC will be demonstratable after implementation.
On the charging infrastructure hardware level, a limitation and timewise modification of provided energy is implemented.
Qwello's charger have no V2G functionality, so the demo site infrastructure will not be used for testing the SMAC extension for this purpose.
In addition to the integration of SMAC in Berlin, IKEM is working together with a local grid operator Stromnetz Berlin GmbH on a publication in connection with SMAC. In the focus of this so called 'proof of concept' is the assessment of barriers and benefits of smart metering solutions and their link to the USER-CHI product. The tasks to processed include the limits of smart metering technology, the connection to roaming platforms, backend and measurement and the assessment of key cost parameters within the investment and operating costs of smart metering solutions compared to the standard measurement methods, as well as explanation of the barriers to providing this data from the point of view of a DSO.

The basis required for this is formed by the obstacles asked about the implementation of SMAC and the findings from the pilot site integration of all cities, which will be evaluate after this

deliverable. In conclusion, the working process is still ongoing. The outcome of this subcontract will be published in summer 2022. As a result, the publication will benefit the pilot sites, the evaluation process of SMAC and supports the scientific discourse and learn process of USER-CHI.

6.3 Budapest

In Budapest the overall sustainable mobility strategy foresees an improvement in the modal split in terms of a major shift from private motorised vehicles to both active mobility and public transport. As reported by BKK (LTP of BUD) 2021 Q3 statistics, the Evs fleet in Budapest is composed by around 16,000 Evs (with a 45% share of fully electric BEVs), while the national EV-fleet numbered 38,000 cars. It is important to note that – together with the Evs registered in the Budapest Metropolitan Area – there are 25,000 electric vehicles circulating in the capital. Moreover, in 2021 Q1, 555 of the 1,471 publicly available recharging stations (with one or two charging points) that have received a licence from the Hungarian Energy and Public Utility Authority (MEKH) operated in Budapest.

From the recharging infrastructure perspective, it is worth noting that in 2020 the electricity consumption in the public network in Hungary reached 7.1GWh (out of the national electricity consumption of 45 TWh), despite COVID. In addition, it has been recorded an increase of 37% concerning the share of DC charging among all 709,579 charging transactions and 49% of electricity transfers were made at high power DC charging stations.

The Hungarian e-mobility laws have defined the set of supporting policies towards a major uptake of zero-emissions vehicles. In particular, the regulation on e-mobility services assumes that the operation of electric charging stations normally comprises also the supply of electromobility services. Nevertheless, operators can reach an agreement with any registered electromobility service provider regarding the supply of e-mobility services to foster the market development and promote the competition among service providers. This can also imply that an electromobility service provider is not obliged to operate chargers and have a corresponding operation license.

State provided resources have not become available for the Budapest for deploying 1100 e-charging points, as planned since 2017. Charging point deployment and operation are taken over by private and other public market players over time, meaning that the Municipality of Budapest haven't installed any e-charging point in the city. The changing regulatory environment and heavy market competition in terms of e-charging point installation and operation made clear that charging point installation and operation by municipalities are loss-making activities, coupled with concerns of economic issues caused by the pandemic, since the charging market in Hungary is dominated by big market players, that are very sensitive about any factors that may distort the market.

The Municipality of Budapest (and its LTP BKK) has no (legal) authorisation to operate e-charging points, therefore Budapest planned to select a CPO/EMSP via public procurement process in the project. Finding a technical partner for the demo was more complicated than anticipated.

This means the Municipality/BKK is not in the position to try to enter the e-mobility market in the publicly accessible segment but would like to perform its roles as a regulator (the Municipality and BKK is running the scale-up phase of micromobility points in Budapest currently, the mobility-network approach fits well to USER-CHI's citizen e-mobility station).

The original concept of the Budapest pilot was to use these mobility points as demonstration sites. These mobility points were planned to operate as multimodal transport minihubs where citizens can choose the vehicles best fitting their individual needs to reach their destination. Mobility points had been tested to include bike-sharing and e-scooter-sharing services. The E-Mobility Service Provider and the Charging Point Operator were to be designated in a public procurement process by mid-2021, but Budapest could not launch any tender due to the lack of technical requirements. The involvement of an implementing partner – as CPO and EMSP – has started, BUD and BKK had several discussions with E.ON Hungary. Parallel to this the Budapest project team investigated the most appropriate testing areas and appointed three locations based on different selection criteria (connection to a public transport HUB, near to a residential area, potential HUB for the shared mobility providers.:

- Etele tér

- Nyugati railway station

- Keleti railway station

These demo areas are available for the public, but primarily Budapest would like to support services such as municipal fleets and shared e-mobility service providers (e.g. carsharing), and based on the results at a later stage publicly available e-chargers are planned to be deployed as part of the city's mobility station installation concept and roadmap.

6.4 Rome

The preparation of the demo of Rome for the SMAC integration is structured in two workstreams.

The first workstream is focused on the infrastructure and on the Hardware installation. Identifying the right demo site has requested several iterations with the partners active in Rome. A first site, along the Colombo roadway, could not be pursued due to the time required to have an administrative approval which is not compatible with the framework of the project. Finally the demo site for SMAC testing has been identified inside the Enel X private premises within what will be called a "V2G Hub Lab" in Via Flaminia. The V2G Hub Lab will host four EnelX JuiceBox DC V2G able to reach up to 25 kW and also PV modules with peak power up to 20 kW and Electric Stationary storage 2nd life with up to 20 kW power peak and 40 kWh energy capacity.

The JuiceBox DC are currently under development and their availability is estimated for Q3 2022. Enel X will rely on an existing intercompany framework agreement for the PV modules and verifying whether it can benefit from framework agreements already available for the ESS 2nd life (from tenders already performed).

The civil works and the related connection works are ongoing.

The second workstream is related to the integration of such infrastructure with Enel X platform with SMAC (and so with INCAR). Enel X is currently working in embedding in its CPO management platform V2G variables. Further Enel X has planned OCPI 2.2 implementation, required to integrate the V2G EVSE with SMAC, by Q3 2022. This will affect the kick-off date for field test trials (possible starting from Q4 2022).

Enel X is also clarifying what kind of readings of the PV modules and ESS can be made available to SMAC for smart charging optimization: at the moment these are the PV power output and the State of Charge.

6.5 Turku

Turku with its partners has been slightly behind schedule in the SMAC integration. Finding a technical partner for the demo was more complicated than anticipated. Tendering for the technical partner with OCPI 2.2 support and V2G charging point capabilities started Spring 2021 and in the Autumn 2021 a subcontract with IGL-Technologies Oy was agreed. After that, regular meetings with the subcontractor has been held with the Turku partners to verify that all the SMAC elements are integrated in the Turku demo site.

Implementation with SMAC should be ready end of summer 2022 and both SMAC and INCAR can be tested in Turku demo sites. In Turku demo, all the charging points installed will use SMAC and INCAR tool. Pääskyvuori demo area is now being built and will have combined charging points with solar panels and batteries. SMAC will control charging points relation with own electricity production and charging load.

Other demo area will test V2G chargers in new Turku Energy headquarters in Kupittaa. These vehicle to grid chargers will be the first of its kind in Turku. Kupittaa headquarters are also under construction and will be ready start of the summer 2022.

7. Conclusions

At this report, the work that was carried out in the context of application development and testing during the tasks T4.3 and T4.4 of WP4 has been presented. The overview of the SMAC solution and its features, technical details and end user documentation is provided. Additionally, the methodology adopted for the evaluation and assessment of the operation of the different software components has been hereby presented.

In summary, the results achieved within T4.3 and T4.4 prove that the SMAC implementation is finalised and ready for the initiation of the piloting phase.

This deliverable concludes the work of WP4. On the overall picture of the USER-CHI project, work takes over in WP6, where the developed solutions will be deployed and integrated with demo partner systems, thus demonstrating all the SMAC features in real environment. Results of T4.4 show that SMAC is in position to start the piloting phase with the appropriate reliability.

Acronyms

Acronym	Meaning
CPO	Charge Point Operator
DSO	Distribution System Operator
EV	Electric Vehicle
GUI	Graphical User Interface
OCPI	Open Charge Point Interface
OCPP	Open Charge Point Protocol
PV	Photovoltaics
RES	Renewable Energy Source
SCSP	Smart Charging Service Provider
SoC	State of Charge
UC	Use Case
UI	User Interface
V2G	Vehicle to Grid

References

[1] USER-CHI, «D4.2- Smart grid services design».

[2] "Open Charge Point Interface 2.2," [Online]. Available: https://evroaming.org/app/uploads/2019/10/OCPI-2.2.pdf.

[3] "Prophet," [Online]. Available: https://facebook.github.io/prophet/.

[4] "Forecasting at scale," [Online]. Available: https://peerj.com/preprints/3190/.

[5] USER-CHI, «D3.3 INCAR - Interoperability, Charging and Parking Platform development and lab-testing».

[6] "NOBEL GRID Project," [Online]. Available: https://nobelgrid.eu/.

[7] "WiseGRID Project," [Online]. Available: https://www.wisegrid.eu/.

[8] USER-CHI Consortium, "D1.1 User requirements for USER-CHI solutions".