Title:	Document Version:
D10.2 WiseGRID Flexibility-based DR Optimization Framework Specification	1.0

Project Number:	Project Acronym:	Project Title:
H2020-731205	WiseGRID	Wide scale demonstration of Integrated Solutions for European Smart Grid

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type*-Security*:
M18 (April 2018)	M18 (April 2018)	R-PU

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other.

**Security Class: PU: Public; PP: Restricted to other programme participants (including the Commission); RE: Restricted to a group defined by the consortium (including the Commission); CO: Confidential, only for members of the consortium (including the Commission).

Responsible:	Organisation:	Contributing WP:
Antonis Papanikolaou	HYPERTECH	WP10

Authors (organisation):
Benjamin Kraft (VS), Jorge Sanjuán, Xavier Benavides, Ignacio Benítez (AMP), Costas Courcoubetis, Antonis Dimakis, Michalis Kanakakis (AUEB), Konstantinos Kompos, Panagiotis Andriopoulos, Antonis Papanikolaou, Thomas Papapolyzos, Kosmas Petridis (HYP)

Abstract:

This task will design demand flexibility models and their behaviour when subject to demand response events alongside a demand response optimization framework to support the business roles of the Aggregator and the Retailer. More specifically, this deliverable's purpose is the design and specification of the WiseGRID DR Optimization framework. Comfort-based demand flexibility models will be developed when environment conditions are available at high-granularity. Also, in lack of low-level information, high-level Demand Elasticity Models will be developed reflecting real-time demand elasticity as a function of multiple environmental and market (price) variables. Lastly, Electric Vehicles Demand Flexibility models will be developed.

Keywords:

Demand response, demand flexibility, thermal comfort, visual comfort, residential and tertiary buildings, demand elasticity

Revision History

Revision	Date	Description	Author (Organisation)
V0.1	14.07.2017	ToC proposal for partner feedback	HYP
V0.2	21.11.2017	Extension of ToC and contents in Section 5	HYP
V0.4	10.03.2018	Drafting of Chapters 3 & 4	HYP
V0.5	16.03.2018	Contribution of EV flexibility model	AMP
V0.6	27.03.2018	Drafting of Chapters 5.2 & 6	HYP
V0.8	17.04.2018	Released deliverable for review	HYP
V0.9	21.04.2018	Feedback from reviewers	AMP
V1.0	24.04.2018	Released for submission	HYP

LIST OF FIGURES

LIST OF TABLES

1 Executive Summary

The goal of the document is the design and specification of WiseGRID DR Optimization framework, as the back-end application running to support the Aggregator and Retailer business roles. More specifically, the purpose of this deliverable is the design and specification of load models and their behaviour when subject to demand response events. To this end, we describe a demand modelling approach along with demand response optimization of loads that takes into account the potential of assets for demand modification in order to participate in alternative demand response strategies for both network operation and market participation optimization.

- Therefore, the first objective is to provide the back-end systems that take into account data as retrieved from the different end-points (sensor data for environment conditions, devices' control setpoints and sub-metering level data) and perform analytics towards the extraction of meaningful information for the business role of the Aggregator (WiseCORP) and the Retailer (WiseCOOP). Such information is used to perform demand modelling & forecasting and demand flexibility profiling. Three demand flexibility models are defined for this purpose: a) Comfort-based Demand Flexibility model, b) Price-based Demand Elasticity model and c) Electric Vehicle Demand flexibility model.
- The second objective is to define the optimization framework which will encompass the abovementioned models allowing in that way the participation of a selection of assets in demand response campaigns. This necessitates the design and specification of interfaces amongst the load models and WiseGRID tools, so that to accommodate the business requirements of the actors considered in the project (DSO, Aggregator and Retailer).

For the **Comfort-based Demand Flexibility model**, the focus is to build accurate models that take into account information related to events from user behaviour. Such events reflect the user's comfort preferences and are crucial in defining dynamic demand profiles; these events allow the extraction of real-time demand flexibility as a function of time, device operational characteristics, environmental context/ conditions and occupant comfort preferences. The main idea behind this modelling framework is the extraction of DER Flexibility Profiles as a function of contextual conditions and occupant's actions. This model is part of the WiseCORP tool and allows an Aggregator to provide flexibility upon DSO's request.

In lack of low-level context information, the high-level **Price-Based Demand Elasticity model** is deployed, reflecting temporal real-time demand elasticity as a function of environmental and market (price and incentive schemes) variables. These prediction models may be utilized by Retailers for the accurate prediction of their clientele's demand such that to purchase adequate energy in the wholesale market and/or achieve a balanced portfolio.

Lastly, EVs are also modelled by the **Electric Vehicle Demand flexibility model** as assets that can offer a multitude of ancillary services, including flexibility for congestion management, regulation services, secondary and tertiary response, etc. that can in principle deliver similar services as batteries under availability constraints.

One of the main goals of WiseGRID is to deploy advanced Demand Response schemes in order to empower energy users and actors. WiseGRID DR Optimization Framework accommodates the needs of DSOs for flexibility requests via orderly defined and standardized interfaces and messages (WG Cockpit), and also caters for the requirements and obligations of other actors, such as Aggregators and Retailers. The DR optimization framework in WiseGRID serves the needs of the aforementioned actors through two high-level demand response scenarios; namely implicit and explicit demand response.

Implicit demand response scenarios focus on the assumption that demand can be shaved or shifted by deploying variable pricing levels throughout the day. On the other hand, explicit demand response scenarios relate to directly controlling loads inside a building (HVACs, lighting devices, etc.). For the former, one needs to know how elastic each building's demand is to different price levels, while for the latter one requires to know the thermal and visual comfort boundaries of the user to offer flexibility without violating occupant comfort conditions.

2 INTRODUCTION

2.1 PURPOSE OF THE DOCUMENT

The aim of the document is the design and specification of WiseGRID DR Optimization framework, as the back-end application running to support the Aggregator and Retailer business roles. More specifically, the purpose of this deliverable is the design and specification of load models and their behaviour when subject to demand response events. To this end, we describe a demand modelling approach along with demand response optimization of loads that takes into account the demand modification potential of assets in order to participate in alternative demand response strategies with the aim of both network operation and market participation optimization.

The objectives of this deliverable are,

- firstly, to model loads that span three asset classes:
    a) building assets that include systems for heating, cooling, air handling, lighting and other controllable buildings loads,
    b) batteries, irrespective of where they are installed or deployed, that will be handled separately from other loads, and
    c) Electric Vehicles.
- Secondly, to define the optimization framework which will encompass the abovementioned models and allow the participation of a selection of asset classes to explicit and implicit demand response strategies.

With respect to the building asset loads to be modelled, the ones with greater capacity to provide demand flexibility are HVAC and lighting devices; these are more appropriate and favourable with respect to DR capacity. Innovative and well-proven machine learning techniques will be utilized in order to improve the accuracy of the models taking into account information of events emanated from user behaviour and respective comfort preferences, towards defining robust and dynamic demand profiles. This will result to the delivery of the Comfort-based Demand Flexibility model as part of WiseCORP, reflecting real-time demand flexibility as a function of multiple parameters, such as time, device operational characteristics, environmental context/conditions and occupant comfort preferences.

Additionally, in lack of low-level context information, high-level Price-Based Demand Elasticity Profiles will be developed, reflecting temporal real-time demand elasticity as a function of environmental and market (price and incentive schemes) variables. Batteries are also going to be modelled as assets that can offer a multitude of ancillary services, including flexibility for congestion management, regulation services, secondary and tertiary response, etc. Finally, Electric Vehicles (EVs) are going to be considered as moving batteries that can in principle deliver similar services as batteries under availability constraints.

2.2 SCOPE OF THE DOCUMENT

The scope of the document is to report the final version of the WiseGRID DR Optimization framework and the models of the respective classes of assets. Therefore, this deliverable is related to tasks T10.2 and T10.3, which refer to the Consumer-Centric Demand Response profiling and Consumer-Centric Demand Response Optimization Modules, respectively.

To this end, the scope revolves around the configuration of comfort-based flexibility models, the definition of appropriate profiling models in response to variable tariffs and financial incentives, the consideration of batteries as flexibility providers, the definition of behavioural profiles for EVs and, last but not least, the optimization framework that will enable the participation of certain loads to implicit and explicit demand response strategies.

The Comfort-based Demand Flexibility model enables the delivery of Context-Aware Flexibility Profiles, reflecting real-time demand flexibility as a function of multiple parameters, such as time, device operational characteristics, environmental context/conditions and occupant comfort preferences. Towards this direction, the incorporation of occupant profiling data to Distributed Energy Resource (DER) models is considered for the extraction of Context-Aware Flexibility Profiles. More specifically, thermal and visual user preferences are associated with the respective HVAC and lighting DER models for the definition of demand flexibility as a function of low-level building information (time, device operational characteristics, indoor environment, and comfort preferences) and are further made available for explicit Demand Response schemes.

In lack of low-level information, the definition of price-based building profiles will be utilized for the extraction of high-level Price-Based Demand Elasticity Profiles. These will define appropriate profiling models and services for defining and analysing the elasticity of demand in response to variable tariff and financial incentive schemes. Such models are the basis upon which individual consumer profiles will be trained on building-level energy consumption information alongside weather information in order to provide accurate elasticity estimations for participation in implicit Demand Response strategies at the portfolio-level.

Additionally, batteries and EVs play an important role in demand flexibility provision, hence their modelling is directly relevant to this deliverable. We configure appropriate battery models. This task will provide accurate battery models in respect to state of charge and discharging rates and allow the provision of ancillary services to the grid by the Aggregator.

The outcomes from the abovementioned models are further utilized in respect to their scope and the relevant DR strategy they participate in and are made available to the WiseGRID DR optimization framework for portfolio-level and building-level control optimization. The extraction of accurate flexibility profiles will further enable the on-line simulation of different portfolio-level or building-level control strategies towards the selection of the optimal strategy that fits to the business models examined in the project.

2.3 STRUCTURE OF THE DOCUMENT

The structure of the document is as follows:

- Chapter 2 provides an overview of the Deliverable focusing on the purpose and scope of the document and its relation to tasks of the project; in other words, this chapter defines the aim and objectives of the current deliverable and sets the boundaries for the flexibility models and the optimization approach used in WiseGRID;
- Chapter 3 gives a list of applicable requirements for the DR framework in respect to the overall list of project requirements; these are the requirements that are relevant to the DR framework and need to be met in respect to their priority; these requirements set the basis upon which chapter 4 builds;
- Chapter 4 outlines the process that is envisaged by the WiseGRID project in order to facilitate the application of implicit and explicit Demand Response campaigns on the relevant electricity end-users, as per the respective business cases (described in chapter 5); more specifically, chapter 5 defines in more detail the scope of each actor involved in WiseGRID as well as the tools that are relevant to the DR framework;
- Chapter 5 describes the business cases examined in WiseGRID along with the relevant workflows and main assumptions for each actor. The two main Demand Response business cases are presented in this chapter, along with the assumptions and workflow that are envisaged to introduce them in a real-world case. This chapter suggests the need of models for flexibility estimation, which are described in chapter 6;
- Chapter 6 describes the models which will be used for the estimation of the available building-level flexibility and elasticity; these will then be used by the tools described in chapter 6.4.1 to facilitate

the business cases of WiseGRID and to implement the relevant DR campaigns. More particularly, it documents the design and specification of WiseGRID Comfort-based demand flexibility model, the Price-Based Elasticity model when there is lack of low-level information from the building environment, and the Electric Vehicle Demand Flexibility model;

- Chapter 6.4.1 draws from the business cases to define high-level Demand Response scenarios and outlines the WiseGRID demand response optimization approach applied on each scenario;
- Lastly, Chapter 8 concludes this Deliverable and recommends further steps to be performed in the project.

3 APPLICABLE REQUIREMENTS

The ultimate goal is for the technological solutions proposed within WiseGRID to be tested and studied in real applications. Hence, a set of requirements is necessary for the systematic analysis of the conditions under which the technological solutions are expected to operate. The following table is a collection of applicable requirements for the DR framework taken from the overall list of project requirements and are defined as a starting point for the design of the framework (overall list defined in D2.1 [1]).

Requirement ID	Description	Classification	Priority
DRF_003	The user needs to be able to configure the electricity tariff, or connect it with some Public API in case of real-time pricing	Consumer-centric demand response framework	5
DRF_004	Energy Storage should be used in order to provide flexibility to the DR	Consumer-centric demand response framework	5
DRF_005	The system should be compatible with others at the project in order to be able to share information	Consumer-centric demand response framework	5
DRF_006	Different types of demand flexibility profiles will be defined as part of the consumer-centric DR profiling addressing the objectives of the project	Consumer-centric demand response framework	3
DRF_007	The comfort-based demand flexibility profiles should be designed taking into account remote monitoring (and controllable) of building loads examined in the project	Consumer-centric demand response framework	5
DRF_008	As part of comfort-based demand flexibility, we should address comfort profiles associated with the operation of energy-hungry HVAC devices	Consumer-centric demand response framework	4
DRF_009	Towards the extraction of visual comfort profiles, information about luminance levels (luminance sensors) under different operational conditions(lighting device status) is required	Consumer-centric demand response framework	5
DRF_010	Towards the extraction of thermal comfort profiles, information about thermal context (temperature & humidity sensors) under different operational conditions (HVAC device status) is required	Consumer-centric demand response framework	5
DRF_011	Towards the extraction of HVAC demand flexibility profiles, information about operational conditions (HVAC device status) and HVAC energy consumption is required	Consumer-centric demand response framework	5
DRF_012	Towards the extraction of Lighting demand flexibility profiles, information about operational conditions (Lighting device status) and energy consumption is required	Consumer-centric demand response framework	5
DRF_014	The extraction of comfort-based flexibility profiles should be based on accurate DER models	Consumer-centric demand response framework	4
DRF_015	Towards the extraction of comfort-based demand flexibility profiles, information about energy cost (retailer tariffs) is required	Consumer-centric demand response framework	4

DRF_016	Comfort-based demand flexibility profiles shall support the implementation of demand shifting strategies (P2H flexibility profiling extraction)	Consumer-centric demand response framework	4
DRF_017	Comfort-based flexibility profiles should ensure the minimum of occupants disturbance on building environment	Consumer-centric demand response framework	4
DRF_018	Comfort based Flexibility Profiles should be exploited towards the implementation of automated DR strategies	Consumer-centric demand response framework	4
DRF_019	Price based Flexibility Profiles should be defined, reflecting the enrolment of prosumers on price based DR scenarios	Consumer-centric demand response framework	5
DRF_020	High-level Demand Elasticity Profiles should be provided in lack of low level information (device level) information	Consumer-centric demand response framework	3
DRF_021	Towards the extraction of price based flexibility profiles, information about market prices (real-time hourly prices, day-ahead hourly prices, pricing schemes) is required	Consumer-centric demand response framework	4
DRF_022	Towards the extraction of price based flexibility profiles, information about external weather conditions should be available	Consumer-centric demand response framework	5
DRF_023	Towards the extraction of price based flexibility profiles, information about individual consumer consumption is required	Consumer-centric demand response framework	5
DRF_025	A central data management unit should be responsible for capturing real-time and historical information required for the extraction of the different profiling types	Consumer-centric demand response framework	5
DRF_026	Real-time information required for the extraction of (comfort-based, price based) Demand Flexibility profiles, should be available in real-time through an automated way	Consumer-centric demand response framework	5
DRF_027	The consumer-centric DR profiling is running as a standalone service calculating the amount of potential flexibility at each demand side end point	Consumer-centric demand response framework	5
DRF_028	An Advanced Flexibility Analysis component should be designed to provide analytics over demand flexibility providing assets	Consumer-centric demand response framework	4
DRF_029	The Advanced Flexibility Analysis should exploit the results from consumer-centric DR profiling engine	Consumer-centric demand response framework	5
DRF_030	Sample analytics over the streams of flexibility data (aggregation, filtering & clustering) will be supported by the Advanced Flexibility Analysis engine	Consumer-centric demand response framework	5
DRF_031	Input values (capacity, response capability, location, time) will set the configuration parameters for the analytics process	Consumer-centric demand response framework	5
DRF_032	Along with real-time analytics, short term forecasting of demand flexibility should be provided by the Advanced Flexibility Analysis engine	Consumer-centric demand response framework	5
DRF_033	The outcomes of Advanced Flexibility Analysis engine may be available for visualization or to a DSS for DR strategies implementation at consumers level	Consumer-centric demand response framework	5

DRF_034	An Optimization DSS component should be designed to enable the aggregation of multiple consumers to participate in DSM strategies	Consumer-centric demand response framework	5
DRF_035	The Optimization DSS component should be designed to allow for the selection of the appropriate aggregated demand side assets to participate in DR programs	Consumer-centric demand response framework	5
DRF_036	The Optimization DSS component should enable interacting with different grid and market stakeholders requesting demand flexibility for the business services	Consumer-centric demand response framework	5
DRF_037	The Optimization DSS component should take into account the different DR contracts towards the selection of customers to participate in the associated campaigns	Consumer-centric demand response framework	5
DRF_038	The Optimization DSS component should be designed to dispatch the DR signal to the different demand side end points	Consumer-centric demand response framework	5
DRF_039	The Optimization DSS component should be designed to dispatch the associated DR signal by taking into account the DR Contract	Consumer-centric demand response framework	4
DRF_040	The Optimization DSS component should estimate the impact of DR strategies to the active consumers, by taking into account the outcomes from consumer-centric DR profiling engine	Consumer-centric demand response framework	5
GEN_005	WiseGRID must promote a 'level playing field' which does not discriminate between competitors (e.g., suppliers, aggregators) as well as flexibility solutions (e.g., storage, DR, EVs)	General Requirements	5
GEN_006	WiseGRID must make use of existing standards or standards under development to provide easier access to market and the dissemination of the resulting solutions worldwide	General Requirements	3

Table 1 – Table of Requirements

4 DEMAND RESPONSE: THE WISEGRID APPROACH

4.1 CONTEXT

One of the main goals of WiseGRID is to develop advanced Demand Response schemes in order to empower energy users and actors, and at the same time provide flexibility to the distribution grid operators. While WiseGRID accommodates the needs of DSOs for flexibility requests via orderly defined and standardized interfaces and messages (WG Cockpit), it also caters for the requirements and obligations of other actors, such as Aggregators and Retailers. In doing so, the WiseGRID Demand Response approach serves the needs of the aforementioned actors through two high-level demand response scenarios; namely implicit and explicit demand response.

Implicit demand response scenarios focus on the assumption that demand can be shaved or shifted by deploying variable pricing levels throughout the day. On the other hand, explicit demand response scenarios relate to directly controlling loads inside a building (HVACs, lighting devices, etc.). For the former, one needs to know how elastic each building's demand is to different price levels, while for the latter one requires to know the thermal and visual comfort boundaries of the user to offer flexibility without violating occupant comfort conditions.

To this end, WiseGRID aims at configuring dynamic visual and thermal comfort models for end-users (consumers). These are associated with energy consumption and load usage metrics in order to define context-aware demand flexibility profiles, as the basis for the definition of optimized flexibility-based Demand Response strategies for the **Aggregator** (explicit Demand Response). The aforementioned models are amongst the tools available in the WiseCORP platform.

Additionally, WiseGRID targets at configuring appropriate elasticity profiling models and services for defining and analyzing the elasticity of demand in response to variable tariff and financial incentive schemes. Such models are the basis upon which individual consumer profiles are trained to provide accurate elasticity estimations so that to accommodate the needs of the **Retailer** (implicit Demand Response) through the WiseCOOP platform.

Lastly, the WiseGRID tool ecosystem includes the WG STaaS/VPP (batteries) and WiseEVP (Electric Vehicles) platforms that are available to the Aggregator for the provision of flexibility services when requested by the DSO and other agents, targeting a stable and secure grid. These platforms incorporate models that reveal the energy needs of batteries and EVs along with their state of charge and discharging rates for appropriate flexibility provision.

This deliverable provides the specification of the functionalities, models, profiles and algorithms for advanced Demand Response Mechanisms, utilizing different flexibility resources, such as building loads.

The following is a contextual representation of the actors involved and the tools they interact with to fulfil their requirements.

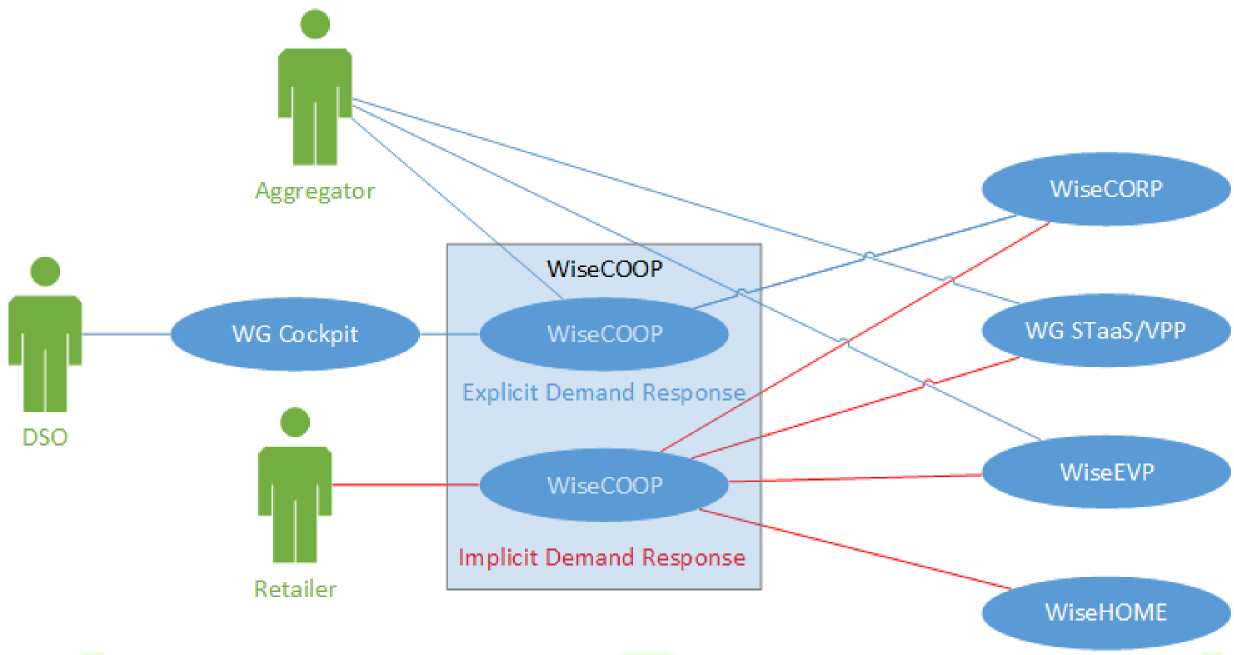

Figure 1 – Contextual representation of Actors and Tools in WiseGRID

4.1.1 Business scenario

The main business scenarios that will be tackled within WiseGRID are the following.

Explicit Demand Response

The DSO interacts using the WG Cockpit in order to request a specific demand modification at certain areas of the distribution network for network management purposes (e.g. congestion relief). This message is broadcasted to the Aggregators through WiseCOOP's explicit demand response interface. The Aggregators in turn, can aggregate available demand flexibility for explicit demand response campaigns from:

- buildings via the collaboration with the facility manager using WiseCORP; this interaction occurs through WiseCOOP's explicit demand response interface with WiseCORP;
- batteries through direct use of WG STaaS/VPP; and
- Electric Vehicles through direct use of WiseEVP.

The Aggregator makes an offer back to the DSO in respect to their available flexibility through WiseCOOP's explicit demand response interface with WG Cockpit.

Implicit Demand Response

Retailers use the WiseCOOP platform to identify imbalances in their portfolio and shape novel pricing schemes to adjust their portfolio's energy consumption. These pricing schemes are determined by the WiseCOOP platform and broadcasted to all WiseGRID platforms.

The following bullets describe the process for the determination, dissemination and use of the prices:

1. WG Cockpit
    - Identifies grid imbalances and network congestion and is controlled by the DSO;
    - The DSO can then request flexibility from Aggregators;
2. WiseCOOP
    - enables Aggregators to manage and implement explicit demand response strategies on buildings;
    - enables Retailers to perform implicit demand response strategies on their portfolio by defining dynamic pricing schemes and communicating them to WiseHOME, WiseCORP, WG STaaS/VPP,

WiseEVP;

3. WiseCORP
    - performs automated (explicit) demand response by scheduling building loads in the near future (intra-day);
4. WG STaaS/VPP
    - enables Aggregators to participate in explicit demand response campaigns using collections of batteries that form a virtual power plant;
5. WiseEVP
    - enables the Aggregators to aggregate Electric Vehicles and provide demand response services upon request of the DSO.

4.2 PROCESS

This subsection outlines the process that is envisaged by the WiseGRID consortium in order to facilitate the concurrent application of both Demand Response campaigns on the same electricity end-users. The main challenge addressed is the fact that explicit DR campaigns must have an assumption about the baseline demand profile in order to establish whether the consumer has actually responded to an explicit DR signal.

As a result, a process is proposed whereby the dynamic prices (implicit DR) are communicated day-ahead to the consumers which provides them the time to respond by optimizing their load scheduling. During the reference day (intra-day), the explicit DR signals are sent and consumers respond by changing the planned load schedules.

The process also describes how the various asset classes will be handled by the different WiseGRID tools.

Main assumptions

During the running day, all field assets (heating, cooling, AHU, lighting, batteries, EVs, etc.) will be divided in three asset classes and each class will be exclusively managed by a single WG aggregator tool with the purpose to deliver as many ancillary services as possible to other grid actors (e.g. DSO, BRP, TSO).

- The three asset classes are:
    - Building assets – systems for heating, cooling, air handling, lighting and other controllable building ***loads***. These will be managed by the WiseCORP application.
        - Building loads can typically deliver demand flexibility (demand shedding of shifting) services for grid congestion management. If managed properly they may also deliver fast response services (because they can shed their demand very fast).
    - Batteries (no matter where they are installed/deployed) will be managed by WG STaaS/VPP
        - Batteries can offer a multitude of ancillary services, including flexibility for congestion management, regulation services, secondary & tertiary response, etc. It makes sense to keep them as a separate asset class that can be optimized for delivering such services
    - Electric Vehicles will be managed by WiseEVP; WiseGRID considered EVs connected on public charging stations.
        - EVs are moving batteries, they can in principle deliver similar services as batteries under availability constraints

Process flow during operation

Day-ahead (day -1) – With price-based/implicit Demand Response and bidding for the day after explicit DR will be operated within this time interval, the following bullet points imply a sequence in time:

1. WiseCOOP obtains wholesale market prices, forecast of 24-hour portfolio demand profile and own generation forecast
    a. WiseCOOP calculates a 24-hour dynamic price for the reference day based on some objective (e.g. portfolio balancing)
        i. It is likely that this dynamic price will be instantiated only as Critical Peak Pricing scheme for the pilot trials, but for purposes of generality it would be good for the tools to be able to handle any dynamic price scheme
2. WiseCOOP broadcasts the price to all interested parties (which should include WiseCORP, WiseEVP and WG STaaS/VPP).
3. WiseCORP will schedule all local building assets (heating/cooling/AHU) as optimally as possible based on the dynamic price in order to first maximise self-consumption and minimize the building level cost for grid imported energy
    - As a result of this optimization, every building asset (HVAC system, AHU, lighting system, etc.) will have its own operation schedule which directly implies an electricity consumption profile for day 0 operation. This schedule will be used as the reference demand profile per device for day 0 management of explicit Demand Response campaigns.
4. WG Cockpit will perform power-flow calculations on its grid and identify potential grid problems.

Intra-day (day 0) – explicit Demand Response campaigns will be operated within this time interval:

1. The three tools (WiseEVP, WG STaaS/VPP & WiseCORP) will continuously collect the available demand flexibility for the following short-term period and calculate the aggregated flexibility potential under their control
    - WiseCORP will deliver a 2-hour demand flexibility forecast, updated every 15/30/60 minutes depending on requirements of other apps
2. If the DSO identifies potential congestion areas, Cockpit will generate a signal to indicate need for demand flexibility (demand shedding or turn-up) with a specific power requirement at specific time periods within the day and at specific network nodes/locations
3. Cockpit will broadcast this signal to any interested party with request for proposal.
4. WiseEVP, WG STaaS/VPP, WiseCOOP will receive the DSO signal
    a) WiseEVP and WG STaaS/VPP analyse it, identify whether the current state of their assets (viz. Electric Vehicles and batteries) allows them to assemble a viable flexibility offer and, an offer is created and sent to the WG Cockpit in response to the flex request. The offer will include the demand flexibility time series to be delivered as well as the expected remuneration;
    b) WiseCOOP asks WiseCORP for the current state of the building assets (devices), assembles a viable bid in respect to DSO's request, bid is created and sent to the WG Cockpit in response to the flex request including potential flexibility per interval to be delivered as well as the expected remuneration;
5. If the offer is accepted by the DSO, the related Aggregator then triggers the DR order to the respective platform (WiseEVP, WG STaaS/VPP, and WiseCOOP).

4.3 ARCHITECTURAL OVERVIEW

In order to gain a better understanding of the scope of the WiseGRID tools related to DR, and the components that comprise them, we initially present a low-level component architecture for WiseCOOP (Figure 2) and WiseCORP (Figure 3), which are the primary platforms responsible for implicit and explicit demand response scenarios, respectively. Thereafter, we provide a high-level overview of the WiseGRID DR tool ecosystem illustrating these scenarios (Figure 4) in order to depict the process followed for each.

An overview of the architecture from the WiseCOOP perspective is given below at the component level.

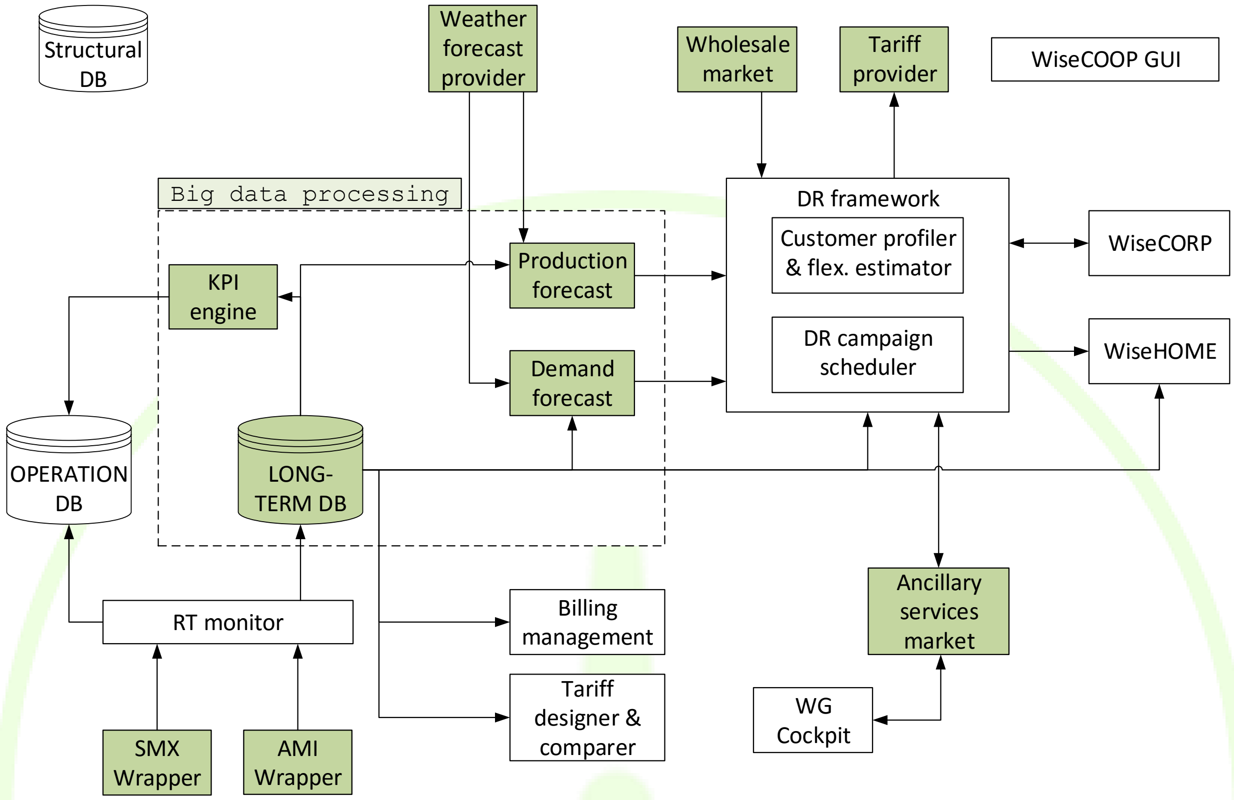

Figure 2 – WiseCOOP platform

In the context of this deliverable, WiseCOOP is responsible for calculating novel pricing schemes for day-ahead and broadcast it to all relevant WG tools. The *Customer profiler & flex. estimator* is responsible for defining appropriate pricing schemes based on inputs from *Production* and *Demand forecast* modules; their use is described in more detail in Chapter 6.4.1. WiseCOOP is also responsible for requesting available demand flexibility from WiseCORP; this is carried out through WG IOP. Lastly, the DR campaign scheduler is responsible for scheduling DR signals and communicating them via WG IOP to other WG tools.

After the DR signal is broadcasted, WiseCORP receives it via WG IOP. In the following figure, a high-level connection of WiseCOOP to the *Energy usage optimizer* component in WiseCORP is shown.

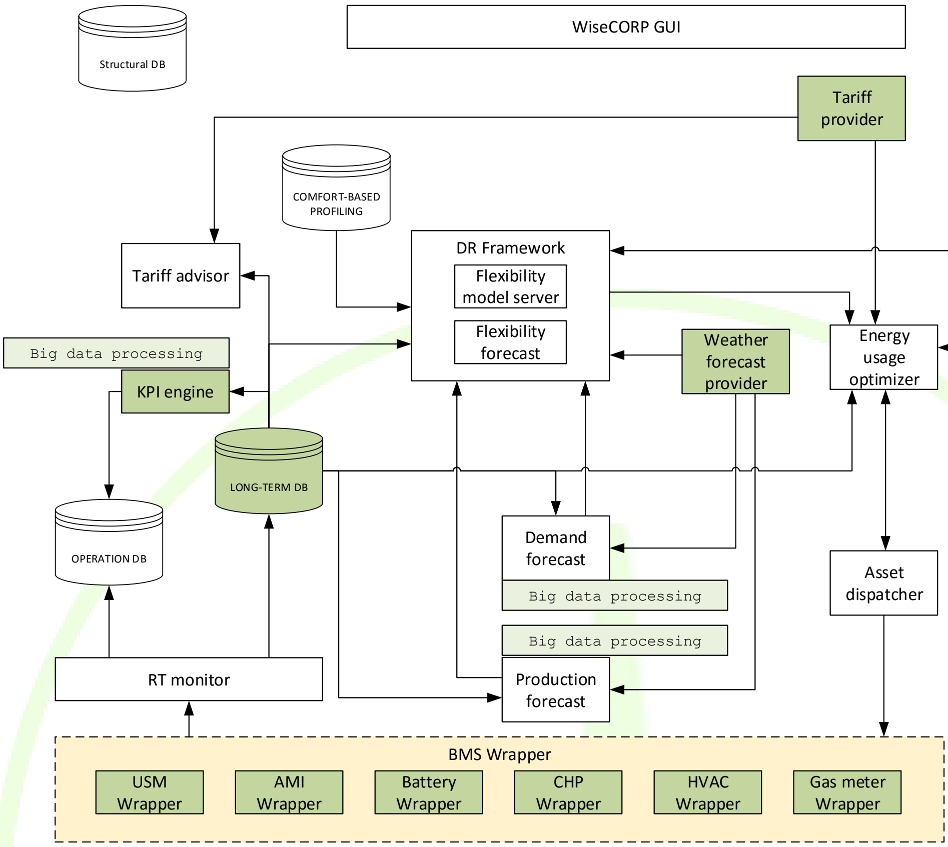

Figure 3 – WiseCORP platform

To put this in perspective, in case an implicit demand response event is triggered, the *energy usage optimizer* is responsible for optimizing energy usage in accordance to energy price per interval broadcasted by WiseCOOP.

All the communications between WiseCORP's components occur via an *internal RabbitMQ service*.

For explicit demand response events, the *Flexibility model server* and *Flexibility forecast* components represent a compacted view of the core tools utilized by WiseCORP in order to estimate real-time potential flexibility and forecast potential flexibility at the device level. These components are shown here in a high-level manner, while they are going to be presented in more detail in chapter 6.2. Control scheduling of devices inside the building is amongst the most important capabilities of WiseCORP in the context of DR scenarios. This is taken care of by the *Asset Dispatcher*, which is responsible for dispatching the appropriate signals on a device-specific manner through the internal rabbitMQ service to the BMS (Building Management System) Wrapper, which, in turn, is responsible for the actual control of the devices in question.

All in all, the following Figure depicts a high-level diagram overview of the entire DR framework/tool defining the high-level boundaries of actors involved and the respective WiseGRID platforms concerned.

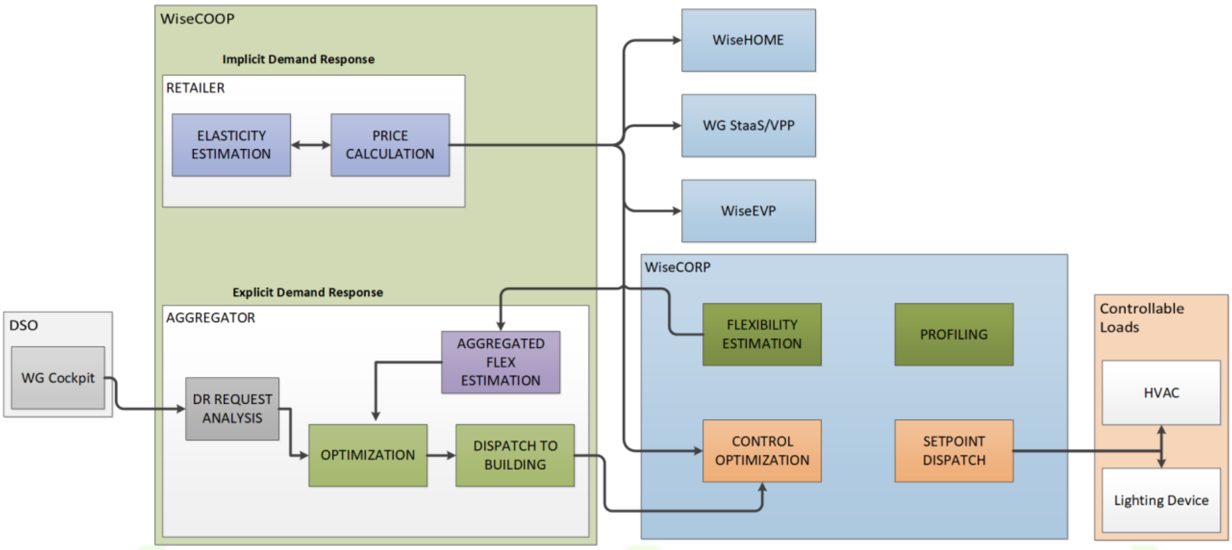

Figure 4 – High-Level Component Level Diagram of the WiseGRID Framework

Figure 4, depicts the process followed (as presented in section 4.2) for each actor and the demand response scenario involved:

1. **Elasticity Estimation & Price Calculation**: this is used by the retailer in order to perform implicit demand response. These tools are used in order to define a dynamic price scheme for day-ahead and communicate it to other WG tools;
2. **DR Request Analysis**: this module represents a high-level view of the message handler within WiseCOOP for negotiation and bidding with DSO; this then forwards the message to other tools in WiseCOOP's explicit demand response framework;
3. **Optimization**: this represents the process that ranks the available buildings in respect to their available flexibility;
4. **Aggregated Flex Estimation**: this represents a high-level view of the component that requests potential flexibility from WiseCORP platform;
5. **Dispatch to Building**: this component dispatches flexibility requests per building;
6. **Control Optimization**: this represents a compacted view of the optimization approach implemented in WiseCORP for translating the flexibility requested to device setpoints;
7. **Flexibility Estimation**: is a high-level view of the flexibility estimation for prospective device setpoints by retaining thermal and visual comfort boundaries;
8. **Profiling**: is the module that performs the user profiling (thermal and visual user comfort profiling);
9. **Setpoint dispatch**: is the component dispatching the output setpoint of control optimization to the available devices.

4.4 INTEGRATION IN THE WISEGRID TOOL ECO-SYSTEM

An illustration of the intended functionality deployment of implicit and explicit Demand Response strategies within WiseGRID, and high-level interfaces with other WiseGRID tools is given below.

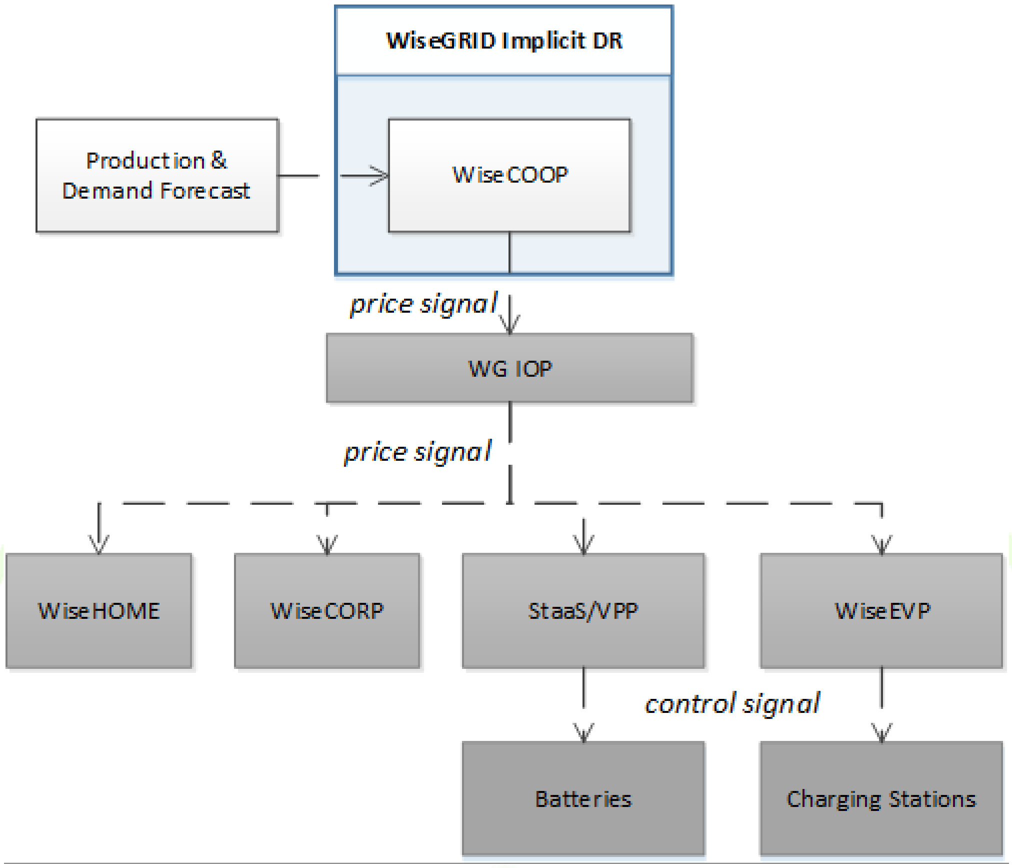

Figure 5 – Day-ahead DR: High-level overview of WiseGRID components and interconnections

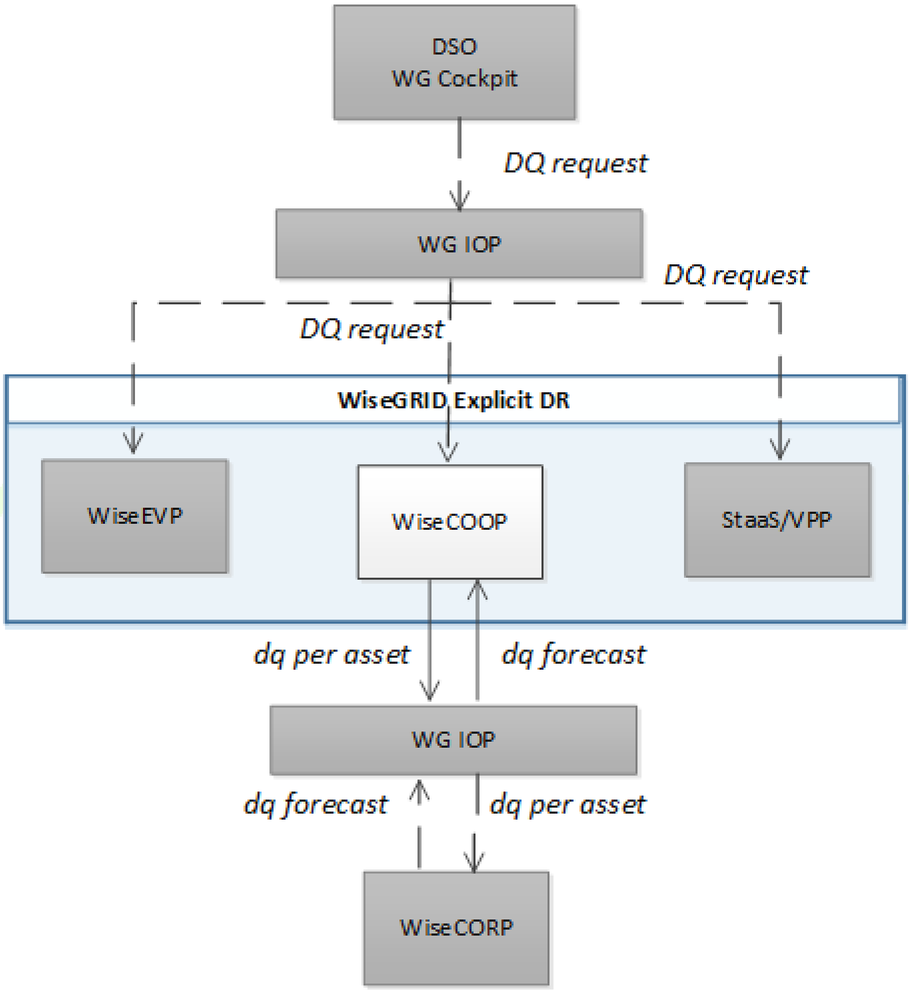

Figure 6 – Explicit DR: High-level overview of WiseGRID components and interconnections

The architectural view of implicit and explicit Demand Response cases shown above reflects the business cases and workflows defined in WiseGRID, which are provided in detail in the following chapter.

5 DR BUSINESS CASES AND WORKFLOWS IN WISEGRID

5.1 IMPLICIT DEMAND RESPONSE

5.1.1 Business case

- Portfolio imbalances and customers: retailer uses the WiseCOOP tool to alleviate portfolio imbalances through dynamic price determination and communication to the energy management component deployed at the premises of electricity consumers (as described in deliverable D7.1 [2])
    - Depending on how often price signals are transmitted to consumers and availability of necessary information, the retailer can calculate the new price (per Program Time Unit – PTU) that can balance its incoming/outgoing energy flows. This is dependent on the information available per pilot site (PTU definition in the various sites).

5.1.2 Main assumptions

- Fairness assumption: all customers receive and are billed according to the same energy price. The main reason lies in the fairness principle that is prevalent in the current electricity retail market, where flexibility is not yet valued by retailers or consumers. Hence, varying prices per customer (including zonal pricing approaches) will not be investigated in the WiseGRID pilot demonstrations.
- Price elasticity: the price elasticity of demand has been estimated and calibrated using correlated energy demand and price time-series. Elasticity will be assumed constant in the relevant price/demand ranges.

5.1.3 Work-flow

- The WiseCOOP tool user interface displays the overlapping forecasts of aggregated portfolio energy demand and total energy supply (including energy purchased in the wholesale markets, local generation, etc.)
    - Demand and supply should be time-series of day-ahead forecasts
    - Time granularity can be 15 minutes. It can be programmable, but actually depends more on the granularity of available time-series or depending on availability per site as mentioned beforehand
    - This graph will immediately indicate to the tool user where the imbalances are in his portfolio
- When the user clicks on each time period where demand/supply are not in balance, he will be lead to a screen where the tool will display the amount of imbalance, the portfolio price elasticity of demand for the specific time period as well as the price differential (positive for excess demand and negative for excess supply) on top of the baseline price that can alleviate the imbalance.
    - The baseline price fluctuation over 24 hours (time-series) is assumed static and known day ahead. Price does not need to be constant throughout the day. What is important is that the day-ahead demand forecast corresponds to the baseline price, so that any resulting price differential can be applied to the baseline price to yield the new price.
- The tool user can accept the new price and this price can be communicated to the consumers

5.1.4 Prerequisites

Time-series of:

- Price elasticity of demand of entire retailer portfolio.
- Day-ahead aggregated demand forecast.
- Day-ahead aggregated local generation forecast.
- Day-ahead forecast of energy purchased in wholesale market.
- Baseline price day-ahead.

5.2 EXPLICIT DEMAND RESPONSE

5.2.1 Business case

The WiseCOOP tool is used by aggregators in order to participate in the market for flexibility based on requests from the network operator or other market actor who requires provision of demand flexibility.

There are two alternative implementation paths that can be followed, depending on the potential interactions between the aggregator and the other market actors.

1. The first case refers to a scenario where multiple aggregators are active in the same (physical or logical) area where flexibility is requested and the total available flexibility (which complies with all the applicable criteria) exceeds the requested one. As a result, each aggregator should coordinate with the DSO (or other flexibility requester) in order to exchange information about his available flexibility and financial offer. The DSO may or may not accept the offer. In the positive case, the aggregator will deliver the flexibility by estimating and dispatching setpoints for the available assets. Using the WiseCOOP interfaces, the aggregator will be able to visualise DR requests (and their specific characteristics) from the DSO (or other flex users), contrast them against the demand and demand flexibility of his own portfolio, estimate the available flexibility that he can offer, send an offer for flexibility provisioning, receive a positive or negative response from the DR requester, calculate the best way (lowest cost, fairness, compliance to contract clauses) to deliver the flexibility, estimate the necessary DR signals and dispatch them to the selected assets.
2. The second case refers to a simpler scenario whereby the aggregator can respond immediately to a DR request, without the need to coordinate with any other actor in advance. This scenario can correspond to a market situation where a single aggregator serves a given (physical or logical) area, hence he has full freedom to satisfy the DR request in the manner he deems optimal.

5.2.2 Main assumptions

- All assets already engaged with the aggregator in a commercial manner are members of the relevant platform, which informs him about the flexibility potential at the appropriate time interval
- No assumption is made on how the assets will actually be controlled, i.e. whether the aggregator will directly control them in a Direct Load Control fashion (automated DR), or whether the aggregator will request a change in their demand profile from the operator of the assets (manual DR). It is evident that the first approach yields more predictable and reliable results, but also places constraints on the availability of tele-command infrastructure.
- Optimizing the selection of assets for participation in DR campaigns relies on a pre-processing of the characteristics in order to select the most promising ones for the specific campaign. This will be facilitated by a clustering methodology that will classify assets into categories ranging from the best assets to utilise in a particular campaign to the worst ones. The classification will be performed based on applicable criteria which can indicatively include their amount of flexibility, location, reliability,

cost, etc. Clustering of important energy metrics per consumer (metrics include energy consumption, demand flexibility in absolute magnitudes, customer remuneration, fairness, etc.) will be performed independently on different time periods during the day. This quantization of time enables the aggregator to focus on the desired time period and assess the benefits of the various assets at the period of interest. This eliminates the chance that asset characteristics that do not apply in the specific period are taken into account (e.g. a home with significant flexibility during the evening cannot deliver any during mid-day when all occupants are away and all important loads are switched off). Per time period the clusters will probably comprise different assets.

5.2.3 Work-flow

- The WiseCOOP tool for the aggregator includes visualizations where demand response requests from DSOs are triggered as a time-series including all relevant information (e.g. relevant grid area)
- In a second diagram in the same screen, the time-series of aggregated flexibility of all assets under the aggregator's control is displayed
- When the aggregator clicks on a specific DR request, he is led to a new screen which displays details of the specific request (power, time period, area). The second diagram changes to illustrate the aggregated flexibility of the assets that fulfil the request constraints (e.g. grid area). The assets are clustered according to relevant criteria (potential flexibility, or other user selectable criterion from a pre-defined list).
- The tool user selects the assets (or the tool automatically selects and proposes the most suitable assets based on a given objective function) that will be involved in the delivery of the desired flexibility in response to the specific request. He selects as many assets as necessary or accepts the tool suggestion. The cumulative flexibility of selected assets as well as the remaining flexibility in order to cover the request requirements are shown in the user interface.
- When a set of assets that fulfils all request requirements is selected, the tool estimates the corresponding DR signals.
- When the user clicks on a "Dispatch" button, the signals are dispatched to the assets and their demand profile is modified. Alternatively, the tool user can specify to the tool at which time in the future to dispatch the signals in order to schedule a flex delivery campaign. The tool will dispatch the signals at the indicated time.

5.2.4 Prerequisites

- The WG Cockpit should generate Demand Response requests and distribute them via the WG IOP, eventually they will be received by WiseCOOP.
- Availability of DR requests with all necessary information (power to be reduced/increased, network area, starting time and duration, etc.)
- Definition of areas of the electricity network. Mapping of assets to the various network areas.

6 MODEL DEFINITION AND CALIBRATION

6.1 INTRODUCTION TO FLEXIBILITY/ELASTICITY MODELS OF THE WISEGRID PROJECT

The goal of WiseGRID is to develop advanced Consumers-Centric Demand Response schemes in order to empower energy users and at the same time provide flexibility to the distribution grid operators. In this chapter, we describe the flexibility models relevant to WiseGRID business cases and how they are related to demand response events. For explicit demand response events, flexibility models are described for batteries as well as comfort-based demand flexibility models that are deployed for buildings where low-level information is available and control of devices is possible. For implicit demand response schemes, high-level price-based demand elasticity models are described.

WiseGRID estimates the demand flexibility available at the device level which preserves the visual and thermal comfort boundaries of occupants. These are incorporated in the WiseCORP platform, and are continuously correlated with indoor and outdoor environment conditions, energy consumption and operational characteristics of devices in order to define context-aware demand flexibility profiles. This sets the basis for the definition of optimized flexibility profiles of devices at building level that serve explicit demand response strategies.

Additionally, this deliverable targets to propose appropriate models for price elasticity of demand in order to analyse and characterize the elasticity of demand in response to dynamic tariffs and financial incentive schemes. Such models are the basis upon which individual building profiles are inferred to provide accurate elasticity estimations for implicit demand response strategies.

Lastly, WiseGRID incorporates models that reveal the energy needs of batteries and EVs along with their state of charge and discharging rates for appropriate flexibility provision.

6.2 COMFORT-BASED DEMAND FLEXIBILITY MODEL

6.2.1 Brief description

As mentioned in the introductory section, this task designs comfort models based on occupant behaviour and proposes DER models for each type of device in order to analyse their response when subject to demand response events. This establishes a holistic demand flexibility modelling approach that examines demand-side capability to participate in alternative demand response strategies that aim both at network operation and market participation optimization. The loads with greater capacity to provide energy demand flexibility are modelled; namely, HVAC and lighting devices, as these are the ones more appropriate and favourable with respect to DR capacity.

In this section the focus is on the calculation of accurate models that take into account information related to events from user behaviour. These comfort preferences are crucial in defining robust and dynamic demand profiles, and lead to the delivery of **Context-Aware Load Flexibility Profiles**, reflecting real-time demand flexibility as a function of multiple parameters, such as time, device operational characteristics, environmental context/ conditions and individual/group occupant comfort preferences. **The main idea behind this modelling framework is the extraction of DER Flexibility Profiles as a function of contextual conditions and actions**. Environmental conditions and user preferences are the driving forces that underlie the operation of devices (from building occupants). Towards this direction, these specific building dynamics have to be defined and incorporated as part of the DER Flexibility models.

The role of the WiseCORP Explicit Demand Response framework is to broadcast the control strategies at building level by taking into account occupants' profiles. This chapter focuses on the presentation of model parameters along with the definition of interfaces, while also providing details about the algorithmic framework considered for the extraction of these configuration parameters. This is actually the objective of this section, to provide the overall learning framework (models and algorithms) for the extraction of Context-Aware DER and Flexibility Profiling Engine.

6.2.2 Context-aware flexibility engine specifications

The goal of this section is to give an overview of the algorithmic framework for the **extraction/learning** of the Context-Aware DER and Flexibility Profiling framework. Therefore, the focus of the work is on the presentation of the algorithmic process towards the extraction of modelling parameters, its design and functional specifications.

The design specifications for the different modules that consist of the Context-Aware Flexibility Profiling Engine are further presented in Figure 7.

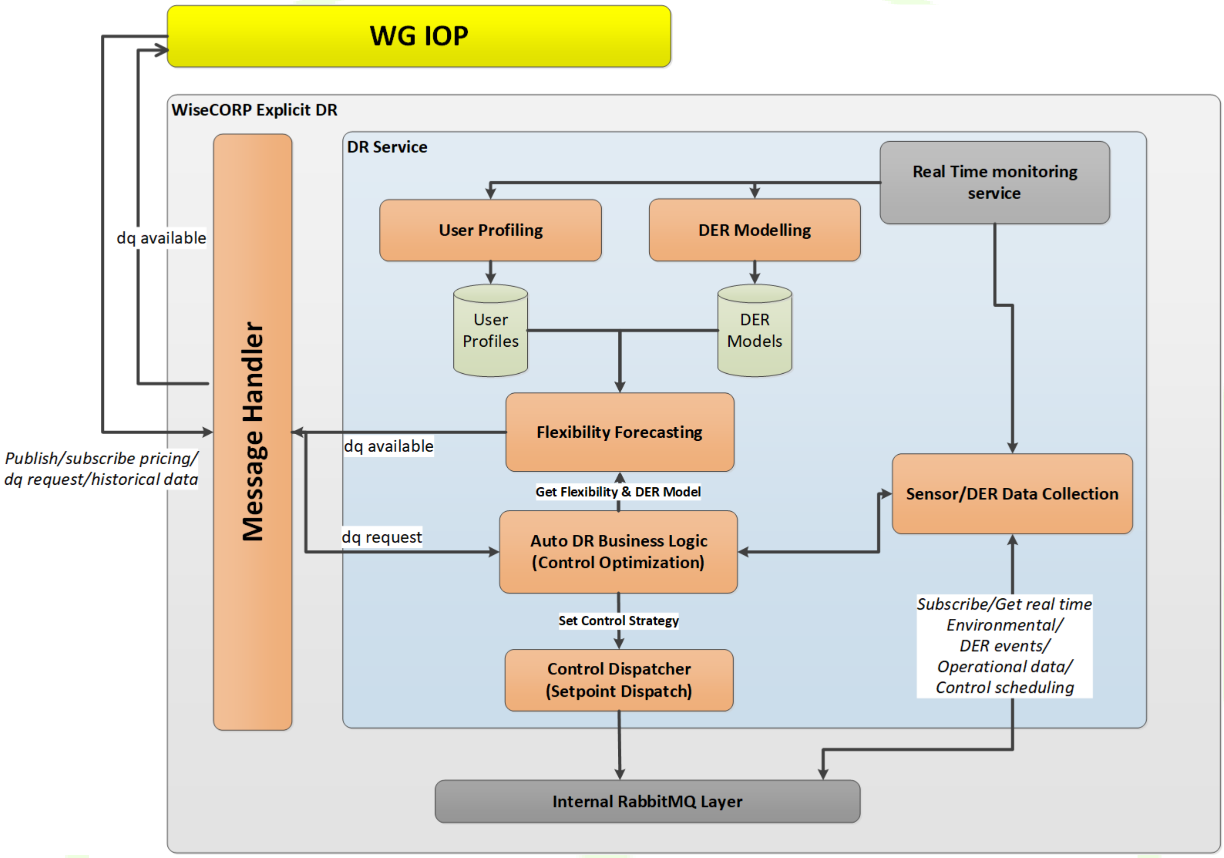

Figure 7 – WiseCORP: Context-Aware Engine Conceptual Architecture

- **Message Handler:** As different types of events will be handled from the component, the role of the Message Handler is to organize the different types of events. More specifically, the module is a subscriber to the following event types:
  a) **Indoor environmental conditions**: namely luminance, temperature, humidity. The structure of the messages exchanged are defined in section 6.2.3;
  b) **External environmental conditions**: namely temperature and humidity as retrieved from available weather services;
  c) **DER operational state** for WiseGRID controllable devices: namely {status, dimming level} for lighting devices, {status, mode, set-point, etc…} for heating/cooling devices, {status} for dual state devices;
  d) **Energy Consumption** data: real-time consumption status of the device.

- **DER Modelling**: various DER models are involved in explicit Demand Response (lighting and HVAC devices); these will be modelled by the respective component. DER models contain the mathematical formulas for the calculation of electric demand (consumption) of each DER type as a function of environmental conditions and configuration parameters affecting DER's operation. These are continuously updated according to the recorded environmental conditions and DER operational data retrieved;
- **User Profiling**: this represents the comfort profiling learning framework and is amongst the core components of the context-aware flexibility framework. It is responsible for learning and periodically updating the user's profile based on **environmental conditions events** and **occupants' control actions;**
- **Flexibility Forecasting:** this module broadcasts the maximum available flexibility as described in section 6.2.8, while it is also utilized by the **Auto DR Business Logic** component for scheduling the setpoints to be dispatched in accordance to a specific DR request;

Prior to the abovementioned analytics process, the message structure and interfaces with sensors and devices is defined, as well as the interface with the Asset Dispatcher. Thereafter, an overview is provided of DER Models along with their respective parameters, Comfort Profiling Learning framework and Flexibility models.

It is clear that raw and processed information is required towards the extraction of accurate Context-Aware Flexibility Profiles. The definition of the aforementioned data types is in line with the overall objective of the profiling engine *(…demand flexibility as a function of multiple parameters, such as time, device operational characteristics, environmental context/ conditions, occupant comfort preferences and health/ hygienic constraints…).*

6.2.3 Sensor and Devices Monitoring and Control Interfaces specification

The following messages are proposed for the communication between the building sensors/devices and the DR framework in order to structure the information exchange, including all necessary meta-data such as timestamps, measurement units, human readable descriptions. Information is encoded using an OBIS-compatible format in order to be easily converted to standardised communication protocols, if needed. The following messages are exchanged through the internal RabbitMQ service that acts as the internal middleware in WiseCORP between devices, sensors and WiseCORP components.

Monitoring of Sensors and Devices

```
"temperature sensor no.1 associated to SHIC01"


=> TOPIC: ASSET01/SHIC01/0-1-96-9-0-1
Document:
{
   "_id" : "0-1-96-9-0-1",
   "assetID" : "assetID",
   "value" : "24.3",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2016-04-15T10:00:00Z"),
   "description" : "Ambient temperature"
}
"luminance sensor no.1 associated to LUX01"


=> TOPIC: ASSET01/LUX01/0-1-151-7-0-1
```

```
Document:
{
   "_id" : "0-1-151-7-0-1",
   "assetID" : "asset01",
   "value" : <latest reading>,
   "unit" : "lux",
   "status" : "1",
   "captureTime" : ISODate("2016-04-15T10:00:00Z"),
   "description" : "Ambient light"
}
"luminance sensor no.1 that PROVIDES TEMPERATURE READINGS (combo)"

=> TOPIC: ASSET01/LUX01/0-1-96-9-0-1
Document:
{
   "_id" : "0-1-96-9-0-1",
   "assetID" : "asset01",
   "value" : "24.3",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2016-04-15T10:00:00Z"),
   "description" : "Ambient temperature"
}

Energy consumption of associated HVAC (SMART-PLUG)

=> TOPIC: ASSET01/SPLUG01/0-1-165-7-0-1
Document:
{
   "_id" : "0-1-165-7-0-1",
   "assetID" : "asset01",
   "value" : [cumulative_active_power, cumulative_reactive_power], → as floats
   "relatedOBIS" : ["1-1-1-8-0-255", "1-1-3-8-0-255"]
   "unit" : "W",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T00:00:00Z"),
   "description" : "smartplug"
}

Energy consumption of associated HVAC (SMART-METER)

=> TOPIC: ASSET01/SLAM01/0-1-165-7-0-1
Document:
{
```

"_id" : "0-1-165-7-0-1",
"assetID" : "asset01",
"value" : [cumulative_active_power, cumulative_active_power], → as floats
“relatedOBIS” : ["1-1-1-8-0-255", "1-1-3-8-0-255"]
"unit" : "W",
"status" : "1",
"captureTime" : ISODate("2017-03-01T00:00:00Z"),
"description" : "smartmeter"
}

Control of IR A/C and LED devices

Command of controlling HVAC devices

TOPIC: ASSET01/SHIC01/0-1-160-7-0-1

Document:

{
"_id" : "0-1-160-7-0-1",
"assetID" : "asset01",
"value" : "23",
"unit" : "grdC",
"status" : "1",
"captureTime" : ISODate("2017-03-01T00:00:00Z"),
"description" : "serial / modbus / IR",
“mode”: “<heating|cooling>”,
"command" : "auto",
"state" : "manual"
}

"command" : "auto" //indicates that the *application* which is identified as “auto” (i.e. auto DR) has requested a change to a new status; for example, “auto” if it is sent by auto DR application, “manual” if it is requested by the user, “other” for other applications (asset dispatcher).

"state" : "manual" // this variable describes the current state (whose application sent the *running* command).

If an off-value is sent (“status” : “0”), the setpoint is ignored.

EXAMPLE request to internal rabbitMQ:

TOPIC: ASSET01/SHIC01/0-1-160-7-0-1

Document:

{
"_id" : "0-1-160-7-0-1",
"assetID" : "asset01",
"value" : "25",
"unit" : "grdC",
"status" : "1",
"captureTime" : ISODate("2017-03-01T00:00:00Z"),

```
   "description" : "IR",
   "mode": "cooling",
   "command" : "auto",
   "state" : "manual"
}
```

EXAMPLE response to internal rabbitMQ:

Content of MQTT message:


```
{
  "_id" : "0-1-160-7-0-1",
   "assetID" : "asset01",
   "value" : "25",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T00:00:00Z"),
   "description" : "IR",
   "mode": "cooling",
  "command" : "", // blank field indicates that the abovementioned message has been executed
   "state" : "auto" // This is the current status of the device (after control command requested)
}
```


Commands for controlling a lighting device

TOPIC: ASSET01/LED0X/0-1-163-7-0-1

Document:


```
{
   "_id" : "0-1-163-7-0-1",
   "assetID" : "asset01",
   "value" : "20",
   "unit" : "%",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T00:00:00Z"),
   "description" : "led lamp",
   "command" : "auto",
   "state" : "manual"
}
```


EXAMPLE request to internal rabbitMQ

Document:


```
{
   "_id" : "0-1-163-7-0-1",
  "assetID" : "asset01",
   "value" : "20",
```

"unit" : "%",
"status" : "1",
"captureTime" : ISODate("2017-03-01T00:00:00Z"),
"description" : "led lamp",
"command" : "auto",
"state" : "manual"
}

EXAMPLE response from internal rabbitMQ.

Document:

{
"_id" : "0-1-163-7-0-1",
"assetID" : "asset01",
"value" : "20",
"unit" : "%",
"status" : "1",
"captureTime" : ISODate("2017-03-01T00:00:00Z"),
"description" : "led lamp",
"command" : "",
"state" : "auto"
}

After presenting the messages exchanged between the internal RabbitMQ implementation and the WiseCORP Explicit DR framework for monitoring and control, the interface between WiseCORP DR framework and Asset Dispatcher is presented.

6.2.4 Interface with Asset Dispatcher

In Section 6.2.3 the structure of messages exchanged between the internal RabbitMQ service and WiseCORP DR framework for monitoring and control of devices and sensors was defined. In this section, the interfacing of WiseCORP DR framework with the Asset Dispatcher is described.

To put this in perspective, in case of an explicit demand response request, WiseCORP DR framework schedules a campaign of setpoints for each device (assets in this context), in order to achieve the requested flexibility request. These are broadcasted in the internal RabbitMQ and, thereafter, the BMS wrapper is responsible for executing the requested signals. However, in order to calculate the flexibility available, the WiseCORP DR framework requires the current and future status of devices which define the baseline consumption. These are provided by the Asset Dispatcher component (as shown in section 4.3).

The *Asset Dispatcher* is responsible for dispatching signal campaigns to the available devices inside the premises and is driven by the *Energy usage optimizer* component; hence, provides setpoint schedules for each device for day-ahead that optimize energy usage of the building. Therefore, an interface between WiseCORP DR framework and the *Asset Dispatcher* is required. In the following table, an example for interfacing with the **Asset Dispatcher** on a publish/subscribe manner via the internal RabbitMQ implementation is presented:

TOPIC : AssetDispatcher

List of object with day-ahead timestamp on a per 15-minute interval for HVAC device:

```
[{
  "_id" : "0-1-160-7-0-1",
   "assetID" : "asset01",
   "value" : "23",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T00:00:00Z"),
   "description" : "serial / modbus / IR",
   “mode”: “<heating|cooling>”,
   "command" : "assetDispatcher",
   "state" : "manual"
},
{
  "_id" : "0-1-160-7-0-1",
   "assetID" : "asset01",
   "value" : "24",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T00:15:00Z"),
   "description" : "serial / modbus / IR",
   “mode”: “<heating|cooling>”,
   "command" : "assetDispatcher",
   "state" : "manual"
}, …,
{
  "_id" : "0-1-160-7-0-1",
   "assetID" : "asset01",
   "value" : "23",
   "unit" : "grdC",
   "status" : "1",
   "captureTime" : ISODate("2017-03-01T23:45:00Z"),
   "description" : "serial / modbus / IR",
   “mode”: “<heating|cooling>”,
   "command" : "assetDispatcher",
   "state" : "manual"
}]
```

"command" : "auto"   //indicates that the *application* which is identified as “auto” (i.e. auto DR) has requested a change to a new status; for example, “auto” if it is sent by auto DR application, “manual” if it is requested by the user, “other” for other applications (asset dispatcher).

"state" : "manual" // this variable describes the current state (whose application sent the *running* command)

List of object with day-ahead timestamp on a per 15-minute interval for lighting device:

```
[{
  "_id" : "0-1-163-7-0-1",
```

```
    "assetID" : "asset01",
    "value" : "20",
    "unit" : "%",
    "status" : "1",
    "captureTime" : ISODate("2017-03-01T00:00:00Z"),
    "description" : "led lamp",
    "command" : "assetDispatcher",
    "state" : "manual"
},
{
   "_id" : "0-1-163-7-0-1",
   "assetID" : "asset01",
    "value" : "20",
    "unit" : "%",
    "status" : "1",
    "captureTime" : ISODate("2017-03-01T00:15:00Z"),
    "description" : "led lamp",
    "command" : "assetDispatcher",
    "state" : "manual"
}, …,
{
   "_id" : "0-1-163-7-0-1",
   "assetID" : "asset01",
    "value" : "20",
    "unit" : "%",
    "status" : "1",
    "captureTime" : ISODate("2017-03-01T23:45:00Z"),
    "description" : "led lamp",
    "command" : "assetDispatcher"
    "state" : "manual"
}]
```


All the necessary monitoring and control interfaces that are required in order to collect real-time indoor environmental conditions and devices' operational characteristics have been defined. In the next section, the configuration parameters required for the definition of DER models deployed in WiseGRID are presented.

6.2.5 DER models configuration parameters

The DER models contain the mathematical formulas for the calculation of electric demand (consumption) of each DER type as a function of dynamic (input data) and static (configuration) parameters affecting DER operation. ANNEX A gives an overview of the required initial configurations for each device at each building.

For example: the DER model for an HVAC system contains the mathematical model that calculates the power consumption of the HVAC given its characteristics (rated power, efficiency, thermal characteristics of the building) and inputs that change dynamically (temperature set point, indoor and outdoor temperature etc.). In addition to energy consumption data, the enhanced DER models defined in the project further incorporate as an output parameter the impact of each DER on indoor environmental condition.

The enhanced model parameters (input/configuration/output) for the demand side device types (lighting, HVAC, plugs) examined in the project are presented next.

Light Devices

Parameter	Description	Units	Type
	Configuration parameters		
type	Type of lighting: incandescent, fluorescent, halogen, CFL and LED.	--	String
Nominal_power	Rated lamp power	W	float
C_{oefF}	The factor expressing the impact of the lamp in the luminance sensor	lux	float
	Input parameters		
status	Current status of the lighting b: 0 (OFF), 1 (ON).	--	boolean
dimming factor	Percentage of the rated power that is being consumed. It is directly linked to the brightness provided by the light even if it is not linearly proportional	%	float
	Output parameters		
Power	Estimated power consumption of lighting	W	float
Luminosity	Estimated luminosity due to lighting device	lux	float

Table 2 – Light Device DER Model

Here an extra configuration parameter (C_{oefF}) which is an outcome of the learning process presented in the next section is defined.

Plug/ Switch Device Model

An ON/OFF plug appliance is any device used that consumes electric energy and the operation is based on the status of the device. An appliance has a rated power, but the real consumption of the appliance depends on the use at any given time. Therefore, the real power that the appliance consumes is given by the utilization factor that depends on the usage of the respective appliance.

Parameter	Description	Units	Type
Configuration parameters			
type	Type of device: printer, home appliance	--	String
Nominal_Power	Nominal power of the office appliance	W	float
Input parameters			
status	Current status of appliance: OFF, ON.	--	Boolean
Utilization factor	Percentage of the nominal power that is being consumed.	%	float
Output parameters			
Power	Estimated power consumption of the office appliance	W	float

Table 3 – Plug/ Switch DER Model

HVAC Device Model

The load type with the highest potential on Demand Response programs is the **HVAC device**. There are two different types of HVAC units examined in the project (ON - OFF heat pumps & Inverter Type Heat Pumps) and thus we need to address the different cases in our models.

The Inverter technology (DC) is the latest evolution of technology concerning the electro motors of the compressors. An Inverter is used to control the speed of the compressor motor, so as to continuously regulate the temperature. The DC Inverter units have a variable-frequency drive that comprises an adjustable electrical inverter to control the speed of the electromotor, which means the compressor and the cooling / heating output. The drive converts the incoming AC current to DC and then through a modulation in an electrical inverter produces current of desired frequency. A microcontroller can sample each ambient air temperature and adjust accordingly the speed of the compressor. The inverter air conditioning units have increased efficiency in contrast to traditional air conditioners, extended life of their parts and the sharp fluctuations in the load are eliminated.

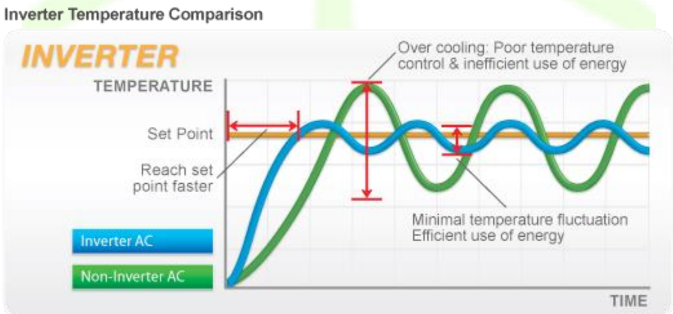

Figure 8 – HVAC set point baseline [3]

In general, it can be said that an inverter air conditioner is always more appropriate than a standard, on-off one. To better understand the difference, let us focus on what happens during the operation, with regards to the energy consumption. An on-off air conditioner absorbs a certain number of watts and then switches off when it reaches the set temperature, returning the consumption or absorption level to zero. An inverter conditioner instead has variable rate of absorption; it adapts to environmental conditions by modulating the energy consumption within a range, varying from minimum to maximum, which goes beyond its nominal power consumption.

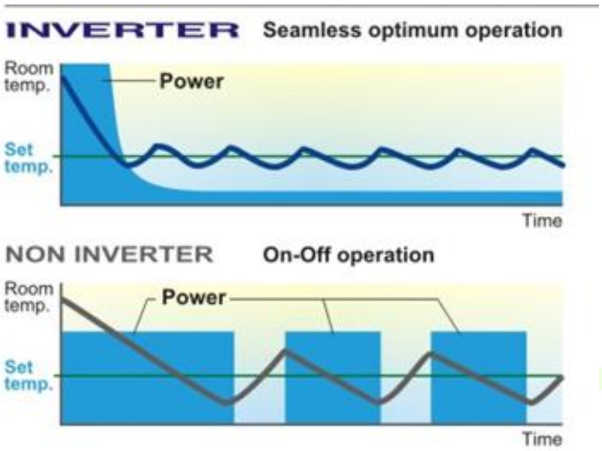

Figure 9 – HVAC energy consumption curve [4]

Within WiseGRID project, where the main objective is the implementation of DR strategies, both types of AC units will be examined. It is not obvious beforehand which is the type of technology in each pilot site, thus we have selected a generic DER model for HVAC devices.

The HVAC DER models the consumption of the system as a function of **heat demand**. The heat demand is therefore calculated, addressing the **thermal model** of each zone examined. The Thermal model contains the thermal characteristics of a building area together with the heat gains produced by the context environment in the area and the desired temperature set point.

Parameter	Description	Units	Type
Configuration parameters			
operation mode	Operation mode of the HVAC system: cooling, heating.	--	String
cooling capacity	Cooling capacity of the HVAC system	W	float
heating capacity	Heating capacity of the HVAC system	W	float
cooling efficiency	Cooling efficiency of the HVAC system (EER)	W_t/W_e	float
heating efficiency	Heating efficiency of the HVAC system (COP)	W_t/W_e	float
Input parameters			
status	Status/Mode/Set-point	--	Complex
heat demand	Heat demand of the thermal zone. The heat demand is calculated in the Thermal Zone model	--	Complex
interior temperature	Internal temperature in the thermal zone at the initial time	ºC	float
Output parameters			
Power	Estimated power consumption of the HVAC system	W	float
final temperature	Estimated final temperature in the thermal zone	ºC	float

Table 4 – HVAC DER Model

A **Thermal Zone** represents the thermal losses and gains of a building area which is controlled by a thermostat. A Thermal Zone contains a set of construction elements describing how the heat is transferred from one area to another area (heat gain due to solar radiation, heat gain due to construction elements, heat gain due to ventilation/infiltrations), also the loads and the occupants of the thermal Zone need to be considered since they act as heat producing elements. All these elements together with the desired temperature set point in the Thermal Zone define the required parameters to obtain the heat demand that is used as input for the HVAC DER.

As this information is not easily retrieved (in real-time), a simplified data driven approach is adopted for modelling heat demand in the project. This is the common case in the literature [5] [6], towards accurately estimating the potential of thermostatically control loads to participate in Demand Response Programmes. The modelling/configuration parameters that specify the heat demand of each building zone are presented in the following table:

Heat Demand Model parameters			
C	Thermal capacitance of the building zone	kWh/ºC	float
R	Thermal resistance of the building zone	ºC/kW	float

Table 5 – Heat Demand Model parameters

Along with the definition of DER model parameters as presented above, the model parameters for demand flexibility profiles should be presented. The proposed framework is considered as context-driven, and thus the incorporation of **human preferences** and **environmental conditions** is a main prerequisite for the definition of flexibility profiling engine. Therefore, the first objective is to define occupants' comfort profiling models that will be further incorporated in the algorithmic process towards the extraction of DER flexibility values.

An overview of the enhanced profiling framework (input/ output/ configuration parameters) is presented here:

Parameter	Description	Units	Type
Configuration parameters			
type	Thermal for HVAC, visual for Lighting	--	String
Device ID	Associated device	ID	ID
Input parameters			
Indoor Env.	Indoor environmental condition	ºC or lux	Float
Output parameters			
Utility Function	Discomfort Level expressed in terms of Utility function	Non dimension	float

Table 6 – Occupants Comfort model parameters

Therefore, the **comfort model** is a **2xN** dimensions table that handles all possible combinations of {indoor environmental conditions → discomfort level of occupants}. Therefore, a non-parametric model (expressed in a tabular format) is adopted for managing the comfort boundaries of the users.

Having clearly defined the different models that consist of the proposed framework, we proceed with the definition of the algorithmic framework (learning process) towards the extraction of these modelling parameters (DER and comfort). This is actually the goal of the work performed in WP10, to provide the analytics engine over the streams of raw data that will enable us to extract accurate DER and comfort profiling model; to be further incorporated for the extraction of the dynamic WiseGRID Context-Aware DER Flexibility profiles. The **algorithmic framework** for the extraction of these profiles is documented in the following sections.

6.2.6 DER models calibration and profiling

Following the presentation of DER model parameters in the previous section, the focus of this section is on the definition of learning framework for the extraction of the configuration parameters. The analysis is again performed first for the DERs and then for the contextual aspects (comfort settings) incorporated in the proposed framework. The analysis covers the different types of demand side load types examined in the project.

Light Device Type

The light device model is defined as: “consumption as a function of status and dimming level”. Therefore, the learning model is based on the definition of the average **consumption values** for the different **device status** and **dimming levels.**

The mathematical formula for lighting device consumption is:

$$P^{output} = Nominal_{Power} * Status * Dimming_{level} \quad (1)$$

A **regression analysis** is performed to correlate input (dimming level & status) and output (consumption) values towards correcting the configuration (Nominal_power) factors. The regression analysis technique is briefly presented.

Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable. A linear regression line has an equation of the form Y = a + bX, where X is the explanatory variable and Y is the dependent variable. The slope of the line is **b**, and **a** is the intercept (the value of y when x = 0).

The most common method for fitting a regression line is the method of least-squares. This method calculates the best-fitting line for the observed data by minimizing the sum of the squares of the vertical deviations from each data point to the line (if a point lies on the fitted line exactly, then its vertical deviation is 0). Therefore, given a random sample from the population, we estimate the population parameters and obtain the sample linear regression model.

Once a regression model has been fit to a group of data, examination of the residuals (the deviations from the fitted line to the observed values) allows the modeller to investigate the validity of assumption that a linear relationship exists. Plotting the residuals on the y-axis against the explanatory variable on the x-axis reveals any possible non-linear relationship among the variables. For the lighting devices, the slope parameter ***b*** is the Nominal_power factor.

The complex part of the learning process is the extraction of lighting device impact on luminance level, (towards the provision of enhanced DER profiles). Caicedo et al. [7] proposed a framework for the disaggregation of illuminance levels on **ambient luminance** and **luminance contribution from lighting devices**. We consider a lighting system in an indoor office, with N light sources and the associated luminance sensors. The average net illuminance w_m at a single (m_{th}) zone in the zone, given that the lighting system is at dimming vector d, may be written as

$$w_m = v_m(d) + u_m \quad (2)$$

Where $v_m(d) = \sum_{n=1}^{M} H_{m,n} d_n$ and u_m are the illuminance contributions due to lighting system and daylight at the m_{th} zone, respectively.

Here, $H_{m,n} > 0$ is the illuminance contribution to the average on the m_{th} zone when the n_{th} light source is at maximum intensity. Denote H to be a matrix whose (m, n)$_{th}$ element is $H_{m,n}$. In most of the installation cases where luminance sensor is installed on the ceiling, Illuminance values at the workspace place however cannot be measured; only illuminance measurements at light sensors are available following the sensor installation as depicted in the next figure (ceiling installation).

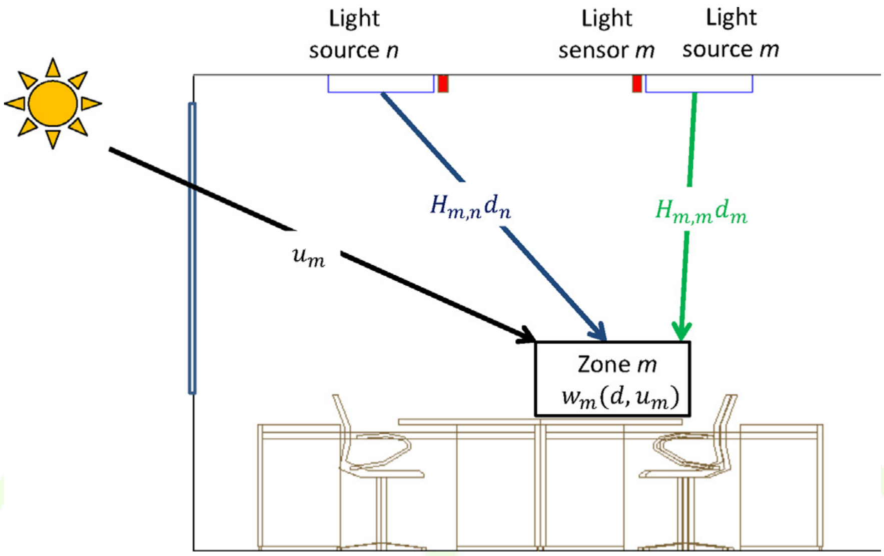

Figure 10 – Illuminance at workspace plane [7]

Therefore, the measured illuminance at a light sensor in the ceiling is the net illuminance due to contributing light sources and daylight **reflected from the objects** (e.g. furniture) in the office. Denote $E_{m,n}$ the measured illuminance at the m_{th} light sensor when the n_{th} light source is at maximum intensity, in the absence of daylight. **We assume that the illuminance scales linearly with the dimming level. This assumption holds well for practical light sources, e.g. LED light sources** [7]. The net illuminance at the m_{th} sensor at the ceiling, given that the lighting system is at dimming vector *d* and under daylight, can then be written as

$$I_m(d, s_m) = \sum_{n=1}^{P} E_{m,n} d_n + s_m \tag{3}$$

Where $\sum_{n=1}^{P} E_{m,n} d_n$ is the illuminance due to the lighting system and s_m is the illuminance due to daylight measured at the m_{th} sensor, as seen in Figure 11. In practice, the mappings $E_{m,n}$ may be computed a priori in a calibration phase by turning on the light sources to the maximum intensity one at a time and measuring illuminance values at the light sensors. Further, we can relate the average illuminance values at the workspace plane and illuminance values at light sensors by

$$\sum_{n=1}^{P} E_{m,n} d_n = \sum_{n} \sum_{p} G_{m,p} H_{p,n} d_n \tag{4}$$

$$s_m = \sum_{p} G_{m,p} u_p \tag{5}$$

Where $G_{m,p} > 0$ is the illuminance contribution at the m_{th} light sensor when the average illuminance at the p_{th} zone due to the n_{th} light source is at the maximum.

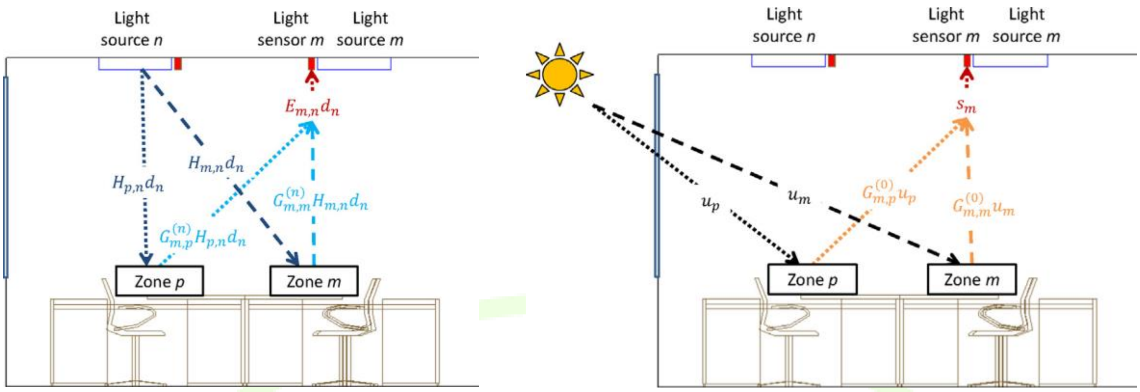

Figure 11 – Illuminance at light sensor [7]

In the WiseGRID case, we define single zones with one luminance sensor installed. Therefore, the parameters of the model presented above are simplified, considering the different controllable lamps and natural sources as the input parameters of the model.

Through analytics over historical data (luminance values and dimming levels) the $G_{1,1}H_{1,n}$ (n-vector, where *n* is the number of lighting devices installed) factor is calculated. This is the **C$_{oeff}$** parameter for each lighting device as presented in the model section above, expressing the **impact context parameter** of a specific lighting device in luminance level. Then, the external luminance level impact is calculated as:

$$s = I(d,s) - \sum_{n=1}^{N} G_{1,1}H_{1,n}d_n \tag{6}$$

Where: $I(d,s)$: luminance sensor values

$\sum_{n=1}^{N} G_{1,1}H_{1,n}d_n$: modelled based calculation of artificial lighting impact to a single lux sensor

To sum up, the lighting DER model parameters are defined by: 1) calculating the **energy profile** as a function of status/dimming level 2) calculating the **impact on luminance level** as a function of status/dimming level, following the algorithmic process as presented above.

Plug/ Switch Device Type

The Plug device model is defined as: "consumption as a function of operational status". Therefore, the learning model consists of the definition of energy consumption for each plug/switch device operational status. The formula for P^{output} calculation is:

$$P^{output} = Nomial\_P \cdot Status \cdot Utilization\_Factor \tag{7}$$

Again, the regression analysis technique is utilized towards correcting the **Nominal_Power** values by analysing:

P^{output} **:** sub-metering information

Status & Util_Factor : plug/switch sensor information

The outcome from the learning process (linear regression analysis) is the **Nominal_Power value** (which in most of the cases is identical to the value defined by the device manufacturer).

HVAC Device Type

This is the most complex device type examined in the project as different parameters affect the HVAC operation performance. While **indoor temperature** data are available from temperature sensor installed in premises, **heat demand** data (as a function of heat gain from solar radiation, occupants, construction elements, ventilation, infiltration) cannot be easily calculated. Therefore, we need to adopt a simplified model that incorporates the **heat demand profiling** parameters as part of the model.

The model proposed in [5] and [6] is adopted towards modelling and controlling thermostatically controlled loads for participation on demand side management strategies.

The dynamic behaviour of the temperature $\theta(t)$ of a thermostatically controlled cooling-load (TCL), in the ON and OFF state and in the absence of noise, can be modelled by:

$$\dot{\theta(t)} \begin{cases} -\frac{1}{CR}(\theta - \theta amb + PR), ON\ State \\ -\frac{1}{CR}(\theta - \theta amb), \qquad OFF\ State \end{cases} \tag{8}$$

Where θamb is the ambient temperature, C is the thermal capacitance, R is the thermal resistance, and P is the power drawn by the TCL when in the ON state.

In steady state, **the cooling period** drives a load from temperature θ_+ to temperature θ_-. Thus solving with initial condition $\theta_0 = \theta_+$ gives

$$\theta(t) = (\theta amb - PR)(1 - e^{-\frac{1}{CR}}) + \theta_+ e^{-\frac{1}{CR}} \tag{9}$$

The same approach is considered for **heating devices** where rated power is $-P$.

Therefore, the calculation of the final temperature is a mixture of input context conditions (indoor air temperature & ambient air temperature) and configuration parameters (C, R, P, set point).

Therefore, the learning process consists of the extraction of C, R parameters that set the thermal demand parameters of each building zone examined. The C parameter further incorporates the time factor, towards simulating HVAC operation through the period of the time. Taking into account HVAC historical data for a short training period, the configuration parameters$C$, R are calculated.

The learning framework for the extraction of the enhanced device models was presented in the previous section. A training period, gathering data from the physical devices is required towards the extraction of the modelling parameters. By having defined these parameters after the learning process 1) model parameters can be periodically updated by taking into account recent data and 2) enhanced real-time DER instances can be provided (by taking into account the input & configuration parameters), further facilitating the DER operation simulation under different contextual conditions.

In the next section, the learning process and the framework of Comfort Profiling engine, incorporated in the project for the extraction of device specific demand flexibility profiles, is presented.

6.2.7 Comfort Profiling Learning Framework

The focus of this section is on the definition of the learning framework for the extraction of Comfort Profiles.

Towards the extraction of comfort profiling parameters, **environmental conditions events** and **occupants' control actions** are incorporated in the learning framework. We review again the data types considered for training purposes (as retrieved from the internal RabbitMQ implementation):

Parameter	Description	Units/Format	Type
_id	The sequence number of sensor	-	ID
assetID	The unique id of the premises	-	ID
value	The metric value of environmental event	Temperature or luminance value	Float
unit	Type of environmental event (Unit type)	[oC or lux]	string
status	Status of sensor (0: inactive, 1: active)	0/1	Integer
captureTime	Time period of the event	YYYY-MM-DD hh:mm:ss	ISODate
description	Description of the environmental event	Ambient Temperature / Ambient Luminance	string

Table 7 – Environmental Event

Parameter	Description	Units/Format	Type
_id	The sequence number of device	-	ID
assetID	The unique id of the premises	-	ID
value	The respective Control Action on the device, triggering change on set-point	Temperature setpoint or dimming level	Float
unit	Type of event (Unit type)	[°C or dimming level]	string
status	The Control Action on the device, triggering change on status (0: close, 1: open)	0/1	Integer
captureTime	Time period of the event	YYYY-MM-DD hh:mm:ss	ISODate
description	Description of the environmental event	Ambient Temperature / Ambient Luminance	String
command	Application triggering setpoint change (or null if no change – monitoring)	auto / manual / other / “”	String
state	Application triggered previous setpoint change	auto / manual / other	String

Table 8 – Control Action Event

Following the high-level taxonomy of events to environmental and control actions, the core part of the comfort profiling framework is the definition of the different types of preferences to be examined in the project. We have already defined the most important comfort models that set the baseline for the WiseGRID framework:

- **Thermal Comfort Profiling**. Thermal comfort is the condition of mind that expresses satisfaction with the thermal environment and is assessed by subjective evaluation (ANSI/ASHRAE Standard) [8]. Maintaining this standard of thermal comfort for occupants of buildings or other enclosures is one of the most important goals of HVAC (heating, ventilation, and air conditioning) design engineers. The Predicted Mean Vote (PMV) model is the main model considered for quantification of comfort level [9]. In WiseGRID, an adaptive model is addressed, considering indoor temperature and HVAC control actions the main parameters of the proposed model adapting in this way in the user preferences.

- **Visual Comfort Profiling**. Visual comfort and discomfort levels of occupants is an obscure concept because of the multitude variables involved and the difficulty of reconciling aesthetic and physiological elements. Most visual discomfort metrics have been derived under controllable conditions in lab environment and represent an average over the subjects, without making any provision for adaptation to individual needs [7] [10] [11]. Therefore, it is mandatory to develop a framework in which visual discomfort can be expressed addressing individual needs and preferences. In WiseGRID framework, we propose a method to calculate a visual comfort probability in a specific zone, relying exclusively on the observation of the users' actions in premises. This user-adaptive approach is delivered as part of the WiseGRID Comfort Profiling framework and defines the visual comfort and discomfort levels of individuals under different luminance conditions.

We select the Bayesian networks as the algorithmic framework for extraction of comfort profiles. The anchor point of the proposed approach is the estimation of **user's discomfort** from a statistical study of his past behaviour. More specifically, Bayes' theorem is applied to estimate a **Bayesian Discomfort Probability** [12] as a function of the temperature/luminance distribution in each building zone. We first set a review of Bayesian statistics and then we discuss how these can be applied in our case.

In statistics, Bayesian inference is a method of inference in which Bayes' rule is used to update the probability estimate for a hypothesis as additional evidence is acquired. Bayesian updating is an important technique throughout statistics, and especially in mathematical statistics. Bayesian inference has found application in a range of fields including science, engineering, philosophy, medicine and law.

A more concrete description of the Bayesian inference follows. Bayesian inference is what we do when we infer that a state A must be true because we have observed state B and that A and B usually happen together. For example, if we see a lion at a circus show we can infer that it must be tame, because all tame lions we have seen were part of a circus show, and we have never seen a wild lion in such a show. Series of experiments have successfully demonstrated that the human brain carries a built-in prior probability curve for different kinds of events, which is updated as new evidence becomes available.

It was Reverend Thomas Bayes (1702–1761) who first discovered what is now known as Bayes' theorem: given two events, denoted by A and B, the following holds:

$$Pr(A|B) = \frac{Pr(B|A)*Pr(A)}{Pr(B)} \tag{10}$$

Where Pr(A) stands for the probability of event A and Pr(A|B) stands for the conditional probability of A knowing that B has happened. Pr(B) can be expanded, yielding the same theorem in another form:

$$\Pr(A|B) = \frac{\Pr(B|A)*\Pr(A)}{\Pr(B|\bar{A})\Pr(\bar{A}) + \Pr(B|A)\Pr(A)} \tag{11}$$

Where Pr(Ā) stands for the probability of A not happening. Bayes' theorem deals with only two events, but Bayesian networks link together an arbitrary number of events believed to exert a probabilistic influence on each other. Consider the following example, adapted from Korb and Nicholson (2003) [13]: a patient's chances of developing lung cancer are assumed to depend exclusively on whether they live in a polluted area, and on whether they smoke. Similarly, having cancer will determine the chances of an X-ray test to be positive and will also affect the chances of the patient developing a breathing condition known as dyspnoea. The probabilistic influences exerted among these events are shown in Figure 12.

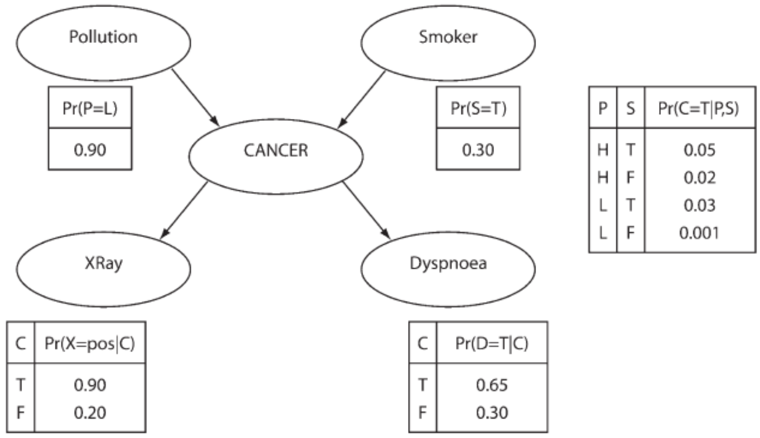

Figure 12 – Bayesian Inference Approach- Reference Example [12]

Here the conditional probabilities are given explicitly, and successive applications of Bayes' theorem allow us to determine any other probability. For example, without knowing whether the patient exhibits dyspnoea and without the results of an X-ray test, the probability of any patient having cancer is:

$$\begin{aligned} \Pr(C=T) = {} & \Pr(\mathrm{C}=\mathrm{T}|\mathrm{P}=\mathrm{H},\mathrm{S}=\mathrm{T})\Pr(\mathrm{P}=\mathrm{H})\Pr(\mathrm{S}=\mathrm{T})+ \\ & \Pr(\mathrm{C}=\mathrm{T}|\mathrm{P}=\mathrm{H},\mathrm{S}=\mathrm{F})\Pr(\mathrm{P}=\mathrm{H})\Pr(\mathrm{S}=\mathrm{F})+ \\ & \Pr(\mathrm{C}=\mathrm{T}|\mathrm{P}=\mathrm{L},\mathrm{S}=\mathrm{T})\Pr(\mathrm{P}=\mathrm{L})\Pr(\mathrm{S}=\mathrm{T})+ \\ & \Pr(\mathrm{C}=\mathrm{T}|\mathrm{P}=\mathrm{L},\mathrm{S}=\mathrm{F})\Pr(\mathrm{P}=\mathrm{L})\Pr(\mathrm{S}=\mathrm{F}) \end{aligned} \tag{12}$$

Where,

- $\Pr(\mathrm{C}=\mathrm{T})$: The probability of cancer.
- $\Pr(\mathrm{P}=\mathrm{H})$: The probability of high air pollution.
- $\Pr(\mathrm{S}=\mathrm{T})$: The probability of being a smoker.

Based on the statistical values as depicted in the schema, we calculate the probability Pr(C = T) = 0.012.

Bayesian inference has emerged in recent years as a particularly promising form of artificial intelligence and has gained a solid foothold in different application domains.

The central point of our claim in the project is that if (even naive) Bayesian classifiers are so good at calculating probabilities, then they should also be able to calculate the probability for a certain environment of being comfortable or uncomfortable to its occupant. Such a classifier should base its judgment on the physical variables it measures and classify the zone examined as comfortable or not. In particular, this classifier will look for correlations between different types of discomfort levels and environmental parameters towards the extraction of accurate behavioural profiles. The principle for the extraction on behavioural profiles based on zone settings and user preferences is provided:

Environment Conditions, Controls & Settings → [Profiling Engine] → Comfort parameters

The next sections presents the project specific implementation of Bayesian networks in the WiseGRID case towards the definition of visual and thermal preference profiles.

Visual Comfort Bayesian networks

As a first step, we continuously record the measured luminance levels, further associated with user actions. If we denote as:

- C: Event "User being comfortable"
- E: Illuminance level
- T: True Indication & F: False Indication as the possible values for C
- e: possible illuminance value for E parameter

We can estimate based on the available data the following parameters:

- $\Pr(E = e|C = F)$, which is the Probability Density Function (PDF) when an abnormal comfort situation is considered.
- $\Pr(E = e|C = T)$, which is the PDF when a normal comfort situation is considered.

If E is a discrete variable we should simply count the number of times it realized each value and divide it by the total number of events. If E is a continuous variable, it is, strictly speaking, a probability density we must estimate. The simplest density estimator is a classic histogram but the choice of bin width can influence the resulting density estimate. Therefore we are selecting a more sophisticated density estimator process named as "taut-string" [14]. This is a nonlinear density estimator, which is locally adaptive, like wavelet estimators, and positive everywhere, without a log- or root transform. This estimator is based on maximizing of a non-parametric log-likelihood function regularized by a total variation penalty.

In Figure 13, an example of the estimated density of illuminance level for a typical building zone when discomfort, $\Pr(E = e|C = F)$ is shown. The data points (users' control) are represented beneath each density curve as small ticks.

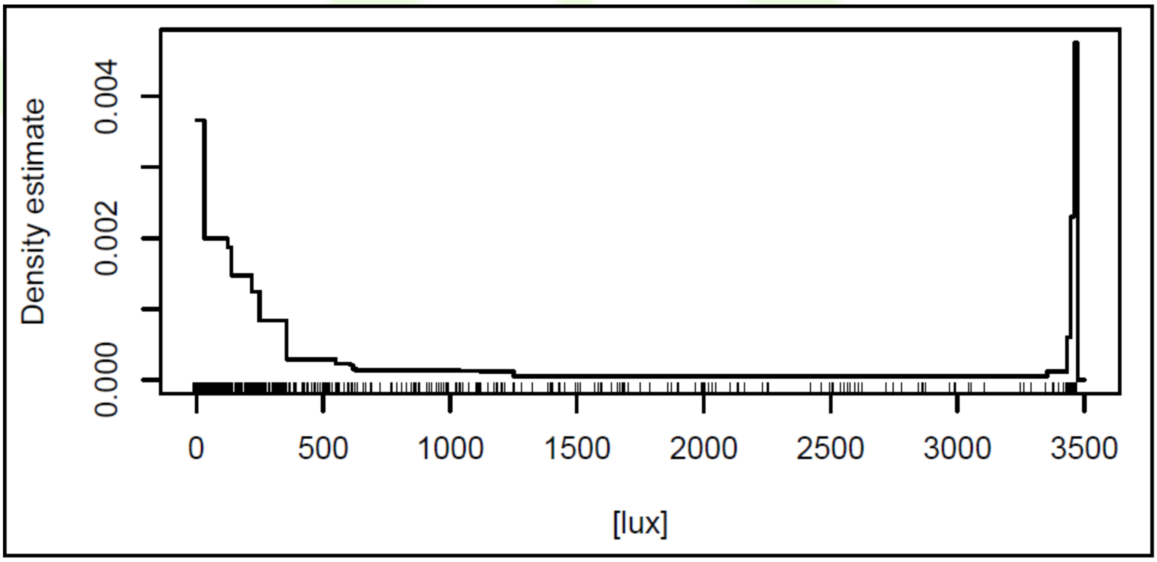

Figure 13 – Probabilistic Density Function for true comfort settings [12]

The users are remarkably consistent in that the illuminance levels most often observed to trigger a user action, for example dimming up the lights, are below about 200 lux, or higher than 3000 lux, where dimming down the lights is more often seen. In other words, only very dark or very bright situations prompt user actions (either these are dimming up or dimming down the lights). Similarly, the distribution of illuminances resulting from user actions tends to cluster around a value of about 400-500 lux. Again, the users are consistent among each other.

From the aforementioned two curves, we may now apply Bayes' theorem and derive Pr(C = False | E = e), i.e. the probability of user discomfort as a function of illuminance level. The estimation of PDF function is depicted:

$$\Pr(\mathrm{C}=\mathrm{F}|\ \mathrm{E}=\mathrm{e}) = \frac{\Pr(\mathrm{E}=\mathrm{e}|\mathrm{C}=\mathrm{F}) * \Pr(\mathrm{C}=\mathrm{F})}{\Pr(\mathrm{E}=\mathrm{e}|\mathrm{C}=\mathrm{F}) * \Pr(\mathrm{C}=\mathrm{F}) + \Pr(\mathrm{E}=\mathrm{e}|\mathrm{C}=\mathrm{T}) * \Pr(\mathrm{C}=\mathrm{T})} \tag{13}$$

The $\Pr(\mathrm{C}=\mathrm{F})$ term, named to Bayesian formalism as the prior, is defined as such that in the absence of any prior information it is safe in most cases to set $\Pr(\mathrm{C}=\mathrm{F}) = \Pr(\mathrm{C}=\mathrm{T}) = 0.5$ [12].

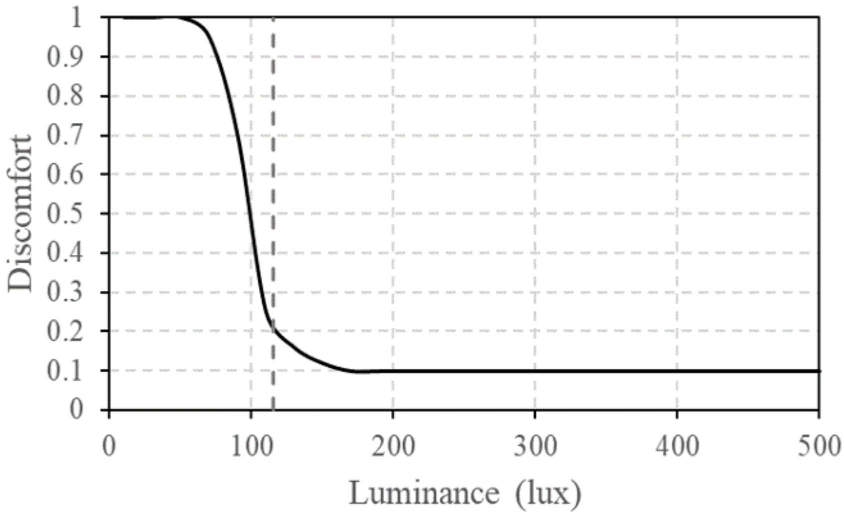

Figure 14 – Visual Discomfort Probability Function

An indicative output curve is shown in Figure 14. A turning point at around 110 lux is depicted showing the maximum comfort/minimum discomfort level. The output curve is assumed to have the above monotonicity with a single boundary showcasing that a user's comfort is decreasing when below the learnt visual comfort threshold, and increasing when above it.

An overview of algorithmic framework for the extraction of visual comfort behavioural profiles is presented in the following:

Input Parameter

Illuminance value (lux: as measured by luminance sensor). User control actions on lighting devices trigger environmental events. Based on correlation of control actions with luminance data, we can define comfort and discomfort visual states.

Algorithmic Framework

Extraction of probability functions: $\Pr(\mathrm{E}=\mathrm{e}|\mathrm{C}=\mathrm{T})$ & $\Pr(\mathrm{E}=\mathrm{e}|\mathrm{C}=\mathrm{F})$ based on available data. Input parameters for probability functions estimation are:

- E: illuminance state
- e: illuminance data
- C: Event "user being comfortable"

Output

The PDF function $\Pr(\mathrm{C}=\mathrm{F}|\ \mathrm{E}=\mathrm{e})$ expresses the discomfort value on a state condition for the single occupant.

Thermal Comfort Bayesian networks

Thermal comfort is defined in terms of the perception of satisfaction that a subject experiences in a given thermal environment. Probably, the most influential standards for designing an indoor environment of thermal comfort have been developed by ASHRAE, the International Organization for Standardisation (ISO) [8], and the European Committee for Standardisation (CEN) [15]. The sensation of thermal comfort is found to be dependent on six environmental and physical factors: air temperature, radiant temperature, air speed, air humidity, and metabolic rate as well as clothing level of the subject. Based on these factors, mathematical expressions of a thermal sensation index, the Predicted Mean Vote (PMV) is delivered, for predicting the percentage of dissatisfied occupants against certain indoor environments were proposed.

The aforementioned model is generic and cannot cover the specificities of each case scenario examined. Therefore, an adaptive thermal approach is proposed to optimise the comfort acceptance of end users. A Bayesian adaptive comfort temperature approach is proposed for WiseGRID needs in order to predict the desired temperature setpoints for an air-conditioned space according to the occupants' complaints about thermal discomfort. In particular, measured system settings and complaint records are used as input parameters to demonstrate the proposed algorithm in determining the optimum temperature set point for the HVAC system.

The main differentiation from luminance framework (as presented above) is that parameter E is a discrete variable. Thus, the thermal discomfort function is extracted as a discrete probability density function and each value will define the utility parameter (Utility function) for thermal preferences. The next table presents the details of algorithmic framework for thermal preferences.

Input Parameter

Indoor Temperature value (as measured by temperature sensor). The user control actions on HVAC units trigger an environmental event. Based on correlation of control actions with indoor temperature conditions we can define comfort and discomfort states.

Algorithmic Approach

Extraction of probability functions: $\Pr(E = e|C = T)$ & $\Pr(E = e|C = F)$ based on available data. Input parameters for probability functions estimation are:

- E: Temperature state
- e: Temperature data
- C: Event "user being comfortable"

Output

The PDF function $\Pr(C = F|\, E = e)$ expresses the discomfort value on a state condition for the single occupant.

By taking into account input data, we infer the thermal profiling curve. Information retrieved from interaction and non-interaction of the users with the system under different environmental conditions, set the datasets further considered in our Bayesian analytics process for the extraction of the associated utility functions.

The overall analysis is performed in a dynamic way, setting different weights on the events examined at the analytics process in order to further address the seasonal aspect of thermal profiling. For the selection of the appropriate weights we take into account the ASHRAE standards for adaptive thermal models (as further adapted in EU with EN 15251: Indoor Environmental Criteria standard).

The adaptive model is based on the idea that outdoor climate influences indoor comfort because humans can adapt to different temperatures during different times of the year. Numerous researchers have conducted field studies worldwide in which they survey building occupants about their thermal comfort while taking simultaneous environmental measurements. These results were incorporated in [8] as the adaptive comfort model. The adaptive charts relate indoor comfort temperature to prevailing outdoor temperature and define zones of 80% and 90% satisfaction. The following figure presents the correlation of outdoor-indoor environmental (temperature) conditions:

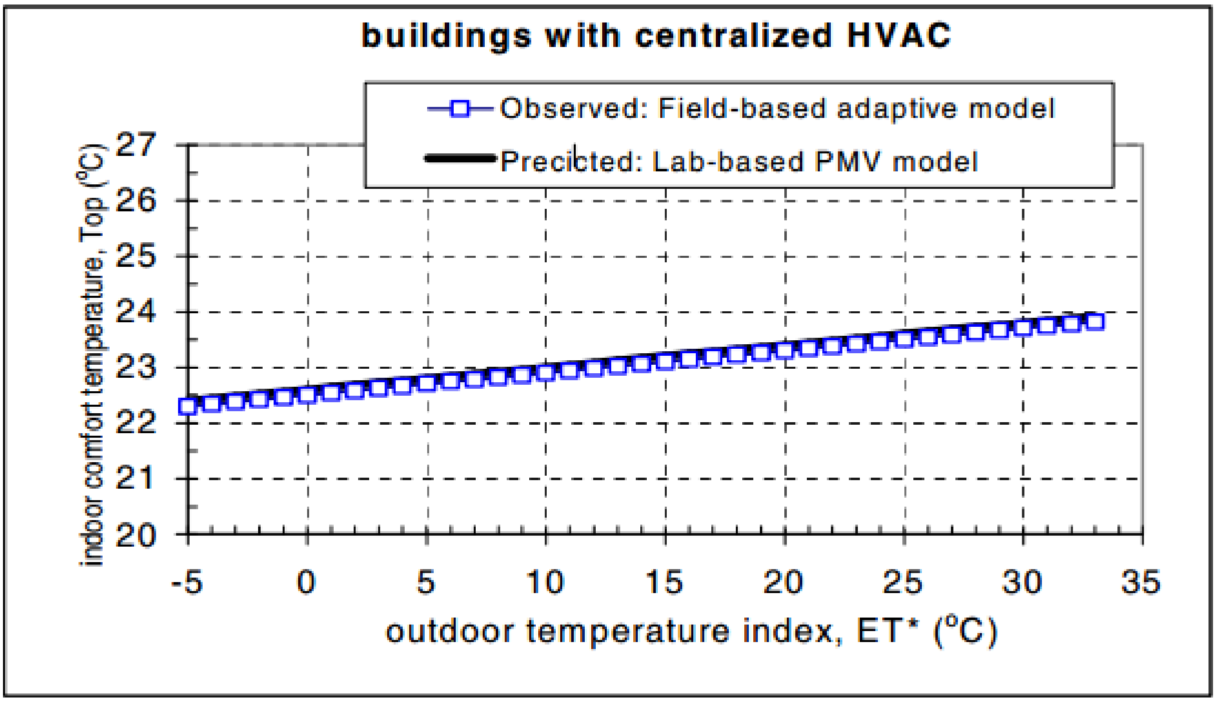

Figure 15 – Observed and predicted indoor comfort temperatures from RP-884 database, for HVAC buildings (ASHRAE) [8]

A pilot site-specific fitted curve is defined in the project, facilitating that way the adaptiveness of the comfort profiling models in the actual environmental conditions.

The aforementioned analysis highlighted the training process for the extraction of comfort profiling model parameters. The outcome of this process is the extraction of thermal and visual (dis)comfort levels (expressed in terms of a Utility function) as a function of internal environmental conditions. Especially for thermal comfort profiles, the analysis is provided in 2-steps, 1) extraction of users comfort levels as a function of environmental conditions and 2) adaptiveness of thermal comfort values by incorporating also outdoor environmental conditions in the analytics process.

An indicative output curve for thermal comfort is shown in Figure 16. In this case, thermal comfort is two-side bounded reflecting the stronger impact that thermal conditions have on the user.

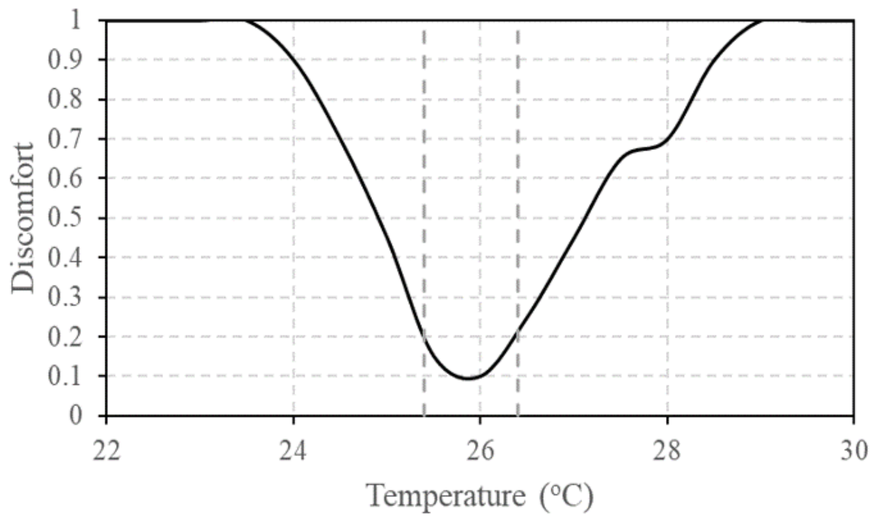

Figure 16 - Thermal Discomfort Probability Function

As a next step of the work, by incorporating this contextual information to the semantically enhanced DER models presented above, the context based demand flexibility profiles can be extracted.

6.2.8 Integrated Demand Flexibility Profiles Framework

Following the extraction of the DER and comfort profiles, the definition of WiseGRID Demand Flexibility Profiling Framework is described. It should be pointed out that there is no static approach that describes the Demand Flexibility profiling framework, rather a dynamic method is adopted for calculating the Demand Flexibility values, taking as input parameters the different demand and comfort profile models as presented above.

DER Flexibility Engine: This is the software module of the WiseGRID Demand Flexibility Engine that incorporates the algorithmic process towards the extraction of demand flexibility profiles. The DER model parameters (**DER Modelling**) further enhanced with (near) real-time environmental and behavioural characteristics (**User Profiling**) enable the extraction of the amount of demand flexibility that each specific device may offer to the WiseGRID system. The data representation of the **DER Flexibility Engine** is presented in the next figure

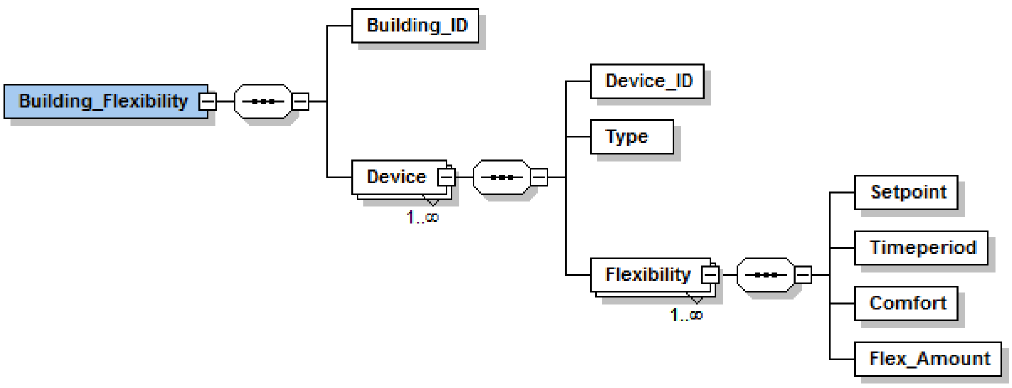

Figure 17 – DER Flexibility Modelling Framework

Where:

- **Set point**: is the operational point of the device,
- **Time-period:** is the associated time-period for the set-point operation
- **Comfort:** is the instance of the comfort profiling mechanism as presented above

- **Flex_Amount:** the amount of demand flexibility associated with the selected **set point** operation.

The algorithmic framework for the extraction of **Flex_Amount** is:

```
% DER Flexibility Modeling Engine
for i=1:Devices
 for j=1:Setpoint
  Actual_Consumption(j)=DER_Model(Device(i), Potential_Setpoint);
  Baseline_Consumption(j)=DER_Model(Device(i), Current_Setpoint);
  Context=DER_Model(Device(i), Setpoint);
  Comfort(j)=Building_Comfort(Device(i),Context);
  Flex_Amount(j)=Baseline_Consumption(j)-Actual_Consumption(j);
 end
end
```

The overall analysis takes into account technical and operational constrains towards the evaluation of different control alternatives. The role of this module is to calculate the potential of controllability of each device type and this information is further available for exploitation at building and portfolio level.

Interfaces specification

The **Auto DR Business Logic** subscribes to **Flexibility Forecasting** to receive the potential of demand flexibility/controllability of the different device types. This is an online service that periodically updates the potential of controllability. (Predefined time-periods: 15 minutes, 30 minutes).

The following message type is available for explicit DR strategies within WiseCORP and at WiseCOOP for the optimization process at building and at portfolio-level, respectively.

Message Type: JSON message with the list of provision of flexibilities for each setpoint, per interval and per asset:


```
[
  {
    "assetKey": "asset01",
    "devices": [
      {
        "deviceId": "0-1-160-7-0-1",
        "deviceType": "HVAC",
        "flexibilityList": [
          {
            "setpoint": 22,
            "interval": 1,
            "comfort": 0.85,
            "flexAmount": 0.120
          },
          {
            "setpoint": 22,
            "interval": 2,
```

```
          "comfort": 0.83,
          "flexAmount": 0.120
        },
        {
          "setpoint": 22,
          "interval": 3,
          "comfort": 0.84,
          "flexAmount": 0.120
        },
        {
          "setpoint": 22,
          "interval": 4,
          "comfort": 0.82,
          "flexAmount": 0.120
        },
        .....
      ]
    },
    {
      "deviceId": "0-1-163-7-0-1",
      "deviceType": "LIGHT",
      "flexibilityList": [
        {
          "setpoint": 0.85,
          "interval": 1,
          "comfort": 0.92,
          "flexAmount": 0.001
        },
        {
          "setpoint": 0.85,
          "interval": 2,
          "comfort": 0.92,
          "flexAmount": 0.001
        },
        {
          "setpoint": 0.85,
          "interval": 3,
          "comfort": 0.92,
          "flexAmount": 0.001
        },
        {
          "setpoint": 0.85,
          "interval": 4,
```

```
            "comfort": 0.92,
            "flexAmount": 0.001
          },
          ....
        ]
      }
    ]
  }
]
```

In addition to the definition of the DER Flexibility Engine that calculates the potential controllability of the different device types, we provide an interface for calculating the maximum potential flexibility available towards the definition of asset participation at portfolio-level. This considers the different business scenarios examined in the project – for example, ranking buildings at portfolio-level that will participate in an explicit DR strategy in response to a DSO's request.

The role of this is to provide an aggregated list of flexibilities for each building, reflecting in that way the maximum available demand flexibility. The main differentiation of this process is that a single demand flexibility potential value is selected as an outcome; this being the aggregated maximum demand flexibility per interval for the building. The algorithmic process for this process is presented:

```
% DER Flexibility Modeling Engine
Total_Flex_Amount=0;
 for i=1:Devices
  Max_Flex_Amount(i)=0;
  for j=1:Setpoint
   Actual_Consumption(j)=DER_Model(Device(i), Setpoint, timeperiod);
   Context=DER_Model(Device(i), Setpoint, timeperiod);
   Comfort(j)=Building_Comfort(Device(i),Context);
   if Comfort(j)> Comfort_Level
     Baseline_Consumption(j)=DER_Model(Device(i), Current_Setpoint, timeperiod);
     Flex_Amount(j)=Baseline_Consumption(j)-Actual_Consumption(j);
     if Flex_Amount(j)> Max_Flex_Amount(i)
        Max_Flex_Amount(i)=Flex_Amount(j);
       Flex_Point(i,j)=(Device(i),Setpoint(j));
      end
   end
   end
Total_Flex_Amount=Total_Flex_Amount+Max_Flex_Amount(i);
end
```

with the following input parameters:

- **Timeperiod**: The requested time period for short term forecasting. (**Default value**: 15 minutes)
- **Comfort_Level Indicator**: A boundary at comfort level expressed in terms of utility function as presented in section 6.2.7. (**Default value:** 0.7)

Therefore, we are incorporating in the analytics process the specific business objectives as defined by the Building Dynamic Assessment Engine. Therefore, we are specifying the interfacing process with the respective system component.

Interfaces specification

This is the dynamic layer of the engine, supporting multiple ad-hoc requests.

Request Message Type defined by the parameters as specified above {Time period, Comfort Level Indicator} and

```
{
  "assetIds": [ "asset01","asset02",...],       // id of the asset or zone
  "comfortLevel": 0.7,          // comfort-level
  "timeperiod":  60             //time ahead of the forecast
}
```

Response Message Type: expressed as a JSON message reporting the **maximum** potential of demand flexibility for the specific case scenario.

```
{
  "assetIdList": [
   {
    "assetId": "asset01",
    "flexPotentialList": [
     {
      "flexibility": 0.256,
      "interval": 1
     },
     {
      "flexibility": 0.131,
      "interval": 2
     },
     {
      "flexibility": 0.131,
      "interval": 3
     },
     {
      "flexibility": 0.131,
      "interval": 4
     }

    ]
   }, ......
```

We presented above the design specifications and interfaces definition for the Context Aware Flexibility Profiling engine.

A short overview of the algorithmic process is provided. A main prerequisite for the estimation of Flex_Amount is the extraction of accurate **DER_Model** & **Building_Comfort** models that will set the functions for the calculation of comfort and energy consumption levels under different DER simulation conditions (**Set-points**). For each controllable device type and potential setpoint derived from the allowed setpoint range of each device, we estimate the actual and baseline consumption and by further taking into account the impact on contextual conditions (and subsequently the comfort levels), we estimate the amount of demand flexibility.

The calculation of the potential amount of demand flexibility is calculated in WiseCORP platform (building level), taking into account the different business simulation scenarios (i.e. device-specific flexibility and maximum available flexibility potential per building). The WiseCORP DR platform is serving the implementation of DR strategies by taking into account real-time estimation of demand flexibility (and thus this service is activated when triggered a DR strategy). In the same time, WiseCORP periodically updates the values of **potential amount** of demand flexibility that each consumer of the portfolio may offer to the Aggregator.

6.2.9 Technical description of application

Following the component functional analysis in the previous section, the **development view** of the respective software component considering components' technical requirements and dependencies, as well as programming languages/technologies that have been used for the development of the associated component is discussed next.

At the application layer, the development of the core application is in Java 8, considering the usage of Spring MVC framework for decoupling the different layers that consist of the application functionalities. As presented above in the components analysis, the development approach ensures the modularity of the application, and further modification of any part of the application without affecting the development process (e.g. the business layer of the application is decoupled from the core analytics part towards the extraction of flexibility profiles).

The minimum hardware requirements for the deployment of the application are an 8-core XEON (or equivalent) CPU, 32GB of RAM, 20TB hard disk and a 64-bit Linux OS. The final deployment of the application (either as a cloud application or hosted in pilot premises) will be defined in WP14 along with the overall deployment of the solution at the different pilot sites, taking into account the scale of the demonstration of the profiling engine.

The detailed design and development view of context based flexibility profiling engine was presented. This is one of the main innovations of the WiseGRID framework towards the implementation of a Demand Response optimization framework that incorporates occupants' behavioural preferences in the decision-making process. Though the development of the context-based flexibility profiling engine is an anchor point of WiseGRID framework, it requires the installation of sensors and controllers in premises. Therefore, in lack of low level information high-level demand elasticity profiles are defined and further integrated in the holistic demand flexibility framework of the project. The high-level demand elasticity component analysis is presented in the following section.

6.3 PRICE-BASED DEMAND ELASTICITY MODEL (AUEB)

6.3.1 Brief description

This section introduces a framework for the implementation of implicit (price-based) DR campaigns. The main part of this work focuses on the profiling of the electricity consumers, i.e., the formulation of their demand with respect to the price and the environmental temperature. More particularly, two cases are considered. The former corresponds to the "Constant Elasticity of Substitution" (CES) model below and assumes that the demand shifts within different periods in a day, in response to different prices (announced by the retailer) and temperatures in those periods. As no inter-period demand shifts are observed by all users, a simpler model is also considered (termed "Simple"), where the consumption during each period depends only on the price and the temperature of this particular period.

In constructing customer profiles based on historical data, the above models are fitted to the data using least-squares linear regression. For the model allowing inter-period demand shifts, two different approaches are employed for data fitting, each giving rise to a different prediction method (the "Normalized" and "Enriched") below. To assess the accuracy of the models, a publicly available energy consumption dataset derived from a dynamic-pricing program at the city of London [16] is used and cross performance evaluation is performed. The findings suggest that by classifying customers into those whose consumption patterns exhibit inter-periods demand shifts and those who do not, and subsequently using the CES model for customers in the first category, and the "Simple" model for customers in the second, improves the accuracy of the models' predictions. Finally, an algorithm is introduced which takes as input the aforementioned prediction methods and computes the dynamic price to be applied such that the collective reaction of the customers results to a desired level of load curtailment.

The prediction models may be utilized by a retailer for the accurate prediction of his clientele's demand such that to purchase the adequate energy in the wholesale market and achieve a balanced portfolio. Even in the case of a portfolio imbalance, the retailer may use the proposed algorithm and apply the suggested dynamic price, avoiding in this way to purchase further energy in the particularly expensive intra-day market.

6.3.2 Utility and demand functions

This section presents the approach for the users' profiling, based on the Constant Elasticity of Substitution model which is commonly employed in energy sector for the prediction of the user's consumption [17], [18], [19]. The model assumes that the elasticity of substitution between pricing periods is constant, even if the actual level of the load varies. Before proceeding it is clarified that the coefficients of the function are user-specific, but in what follows the user indicator is omitted for simplicity reasons.

The Constant Elasticity of Substitution utility function of a consumer on day t with J dynamic pricing periods, is as follows

$$u_t = \left(\sum_{j=1}^{J} a_{j,t}^{\frac{1}{\varepsilon^{sub}}} Q_{j,t}^{\frac{\varepsilon^{sub}-1}{\varepsilon^{sub}}} \right)^{\frac{\varepsilon^{sub}}{\varepsilon^{sub}-1}} \tag{14}$$

Where ε^{sub}, is the elasticity of substitution, $Q_{j,t}$ is the electricity consumed during period j on day t and $a_{j,t}$ is the parameter capturing the impact of the environmental temperature on the user's demand. Using standard manipulation, the Marshallian demand function for electricity consumption in period j on day t is given by:

$$Q_{j,t} = \frac{m_j a_{j,t} P_{j,t}^{-\varepsilon^{sub}}}{P_t^{1-\varepsilon^{sub}}} \tag{15}$$

Where $m_t = \sum_{j=1}^{J} Q_{j,t} P_{j,t}$ is the budget, i.e., the total expenditure for the consumption of electricity on day t, and P_{jt} is the price of electricity in period j and day t, while

$$P_t = \left(\sum_{j=1}^{J} a_{j,t} P_{j,t}^{1-\varepsilon^{sub}} \right)^{\frac{1}{1-\varepsilon^{sub}}} \tag{16}$$

is a consumer price index for day t.

So far, if the values of the parameters $a_{j,t}$, ε^{sub} and , m_t were known, the electricity consumption in period j and day t could be computed from Equation (15), given the price vector applied during all the periods of the day. In the sequel, algebraic transformations and least-squares linear regression are used for the estimation of their values. Before proceeding, it is clarified that the investigated model considers only two periods of consumption within the same day, namely the “peak” and the “off-peak” and notated as “p” and “op” respectively.

Taking logarithms in Equation (15) and subtracting $lnQ_{op,t}$ from $lnQ_{p,t}$ results to:

$$ln\left(\frac{Q_{p,t}}{Q_{op,t}}\right) = lna_{p,t} - lna_{op,t} + \varepsilon^{sub} ln\left(\frac{P_{op,t}}{P_{p,t}}\right) \tag{17}$$

Now assume that the parameter $a_{j,t}$ changes in period $j \in \{p, op\}$ and day t according to the formula,

$$a_{j,t} = \widetilde{a_{j,t}} e^{\beta T_{j,t}} e^{\delta_{j,t}} \tag{18}$$

meaning that it is a function of the environmental temperature ($T_{j,t}$) and the coefficient $\widetilde{a_{j,t}}$ during the specific period j and day t and a random term $\delta_{j,t}$. Then, letting $\pi_{p,op,t} = ln\widetilde{a_{p,t}} - ln\widetilde{a_{op,t}}$ and $\mu_{p,op,t} = \delta_{p,t} - \delta_{op,t}$ we get the estimating equation:

$$ln\left(\frac{Q_{p,t}}{Q_{op,t}}\right) = \pi_{p,op,t} + \varepsilon^{sub} ln\left(\frac{P_{op,t}}{P_{p,t}}\right) + \beta\left(T_{p,t} - T_{op,t}\right) + \mu_{p,op,t} \Longrightarrow$$

$$ln\left(\frac{Q_{p,t}}{Q_{op,t}}\right) = \mu\pi_{p,op,t} + \varepsilon^{sub} ln\left(\frac{P_{op,t}}{P_{p,t}}\right) + \beta\left(T_{p,t} - T_{op,t}\right) \tag{19}$$

Where,

$$\mu\pi_{p,op,t} = \pi_{p,op,t} + \mu_{p,op,t} = ln\widetilde{a_{p,t}} - ln\widetilde{a_{op,t}} + \delta_{p,t} - \delta_{op,t} \tag{20}$$

At this point everything is set to deal with equation (15) using simple algebra. Assuming that m_t is known (Equation (29) presents the approach for its estimation), the only unknown variable is $a_{j,t}$. The combination of Equations (15), (16) and (18) results to the following equation:

$$Q_{j,t} = \frac{m_t \widetilde{a_{j,t}} e^{\beta T_{j,t}} e^{\delta_{j,t}} P_{j,t}^{-\varepsilon^{sub}}}{\left(\sum_{j=1}^{J} \widetilde{a_{j,t}} e^{\beta T_{j,t}} e^{\delta_{j,t}} P_{j,t}^{1-\varepsilon^{sub}}\right)}, j \epsilon \{p, op\} \tag{21}$$

Setting $\widetilde{a_{op,t}} = 1$ and $\delta_{op,t} = 0$ in Equation (20), results to:

$$\widetilde{a_{p,t}} e^{\delta_{p,t}} = e^{\mu\pi_{p,op,t}} \tag{22}$$

Replacing (22) in (21) for two periods of interest concludes to:

$$Q_{op,t} = \frac{m_t e^{\beta T_{op,t}} P_{op,t}^{-\varepsilon^{sub}}}{\left(e^{\beta T_{op,t}} P_{op,t}^{1-\varepsilon^{sub}} + e^{\mu\pi_{p,op,t}}\ e^{\beta T_{p,t}} P_{p,t}^{1-\varepsilon^{sub}}\right)} \tag{23}$$

And

$$Q_{p,t} = \frac{m_t e^{\mu\pi_{p,op,t}}\ e^{\beta T_{p,t}} P_{p,t}^{-\varepsilon^{sub}}}{\left(e^{\beta T_{op,t}} P_{op,t}^{1-\varepsilon^{sub}} + e^{\mu\pi_{p,op,t}} e^{\beta T_{p,t}} P_{p,t}^{1-\varepsilon^{sub}}\right)} \tag{24}$$

Finally, Equation (25) formulates the demand function of period j at day t, assuming that it depends only on the price and the environmental temperature during this specific period.

$$lnQ_{j,t} = \eta_j + \varepsilon_j lnP_{j,t} + \xi_j T_{j,t}, j \in \{p, op\} \tag{25}$$

In this case, η_j is a constant number, ε_j is the user's price elasticity, i.e., the coefficient which captures the impact of the price on the user's demand and ξ_j is the user's weather sensitivity, i.e., the parameter that captures the impact of the environmental temperature on the user's demand.

6.3.3 Dataset

The evaluation of accuracy of the proposed models, is based on a dataset available online [16], which was derived by a dynamic-pricing (Time of Use) program applied to a large number (~1100, 163 of which were employed for the experimental setup) of households in London, during the 2013 calendar year. This set includes half-hourly data for the total consumption of each household, meaning that no sub-metering of each individual device is available. Concerning the dynamic price announced to the consumers, it varies in the set$\{0.0399, 0.672\}$ $pounds/KWh$, while the consumers were exposed for most of the time to its base value which equals 0.1176 pounds/KWh. It should be mentioned beforehand that the dataset includes only a limited number of Time-of-Use events, meaning that the base value of the price has a dominant presence and it rarely changes to its dynamic value. This fact assigns a further difficulty to the performance of the prediction models, apart from the fact that they consider only the price and the temperature as the parameters affecting the demand and exclude other factors such as the income, the members of the household, etc. (which are not available).

Recall that the followed approach in this section considers only two periods of consumption (namely the peak and off-peak). To this end, targeting to convert the dataset into a compatible form for the study, a pre-elaboration of the dataset was carried out (before the prediction models were applied at it). More specifically, the average value of the applied price, the average environmental temperature (outside the household premises) and the aggregate consumption of each household were computed, during the time intervals [17:00-23:00] and [23:00-17:00] for the peak and off-peak periods respectively. For clarity reasons, it is mentioned that the two former parameters are equal for all households (for each specific period and day), while the consumption differs for each end-point according to its residents' reaction to them.

Based on the derived data, in what follows the major difference of the proposed models is presented, compared with the one introduced in [17], which consists to the considered impact of the outside temperature on each household's demand. More specifically, the approach followed in this section has formulated (for all the models) the demand as a function of the outside temperature, while the work in [17] assumes that the demand is affected by the absolute difference between the outside temperature and the one that the residents perceive as comfort-level within their premises. This latter relation is precise if the HVAC devices are used both for cooling and heating, meaning that they consume more electricity as the outside temperature tends towards its extreme values during the summer and the winter respectively. Actually, this formulation of demand for such type of devices is also proposed in a fundamental research work in the energy sector which investigates optimal implementation of implicit DR campaigns [20].

This variation is justified by multiple reasons which are explained next. Firstly, as already mentioned the available dataset does not provide consumption records separately for each device (for the HVAC in particular), and consequently the estimation of the comfort level is practically infeasible. Additionally, the households in London rely mostly on gas for covering their heating needs, while the use of HVAC is generally limited since the temperature during the summer rarely exceeds the humanly tolerable levels. Despite this fact, the temperature is still expected to have an impact on the users' consumption since it affects their lifestyle patterns, e.g., during the summer the citizens tend to spend more time outside of their houses, the needs for lightening is decreasing etc. Figure 18, juxtaposes the aggregate consumption (of all the users) with the environmental temperature, for both periods and for all days of the year. Before proceeding, it is clarified that the off-peak consumption appears greater than the peak one, since it reflects the sum of consumption during all the 18 hours of the period (in hourly average terms, it would be clearly lower). Figure 18 reveals the seasonality of the consumption, having a decreasing tendency with respect to the temperature, a fact indicating that the aforementioned approach for the formulation of the demand function is appropriate for this dataset. It is emphasized that the proposed models may be easily adjusted, targeting to capture the impact of temperature as considered in the related work (described above), if applied at sites where the HVAC equipment is used for covering both the cooling and heating needs.

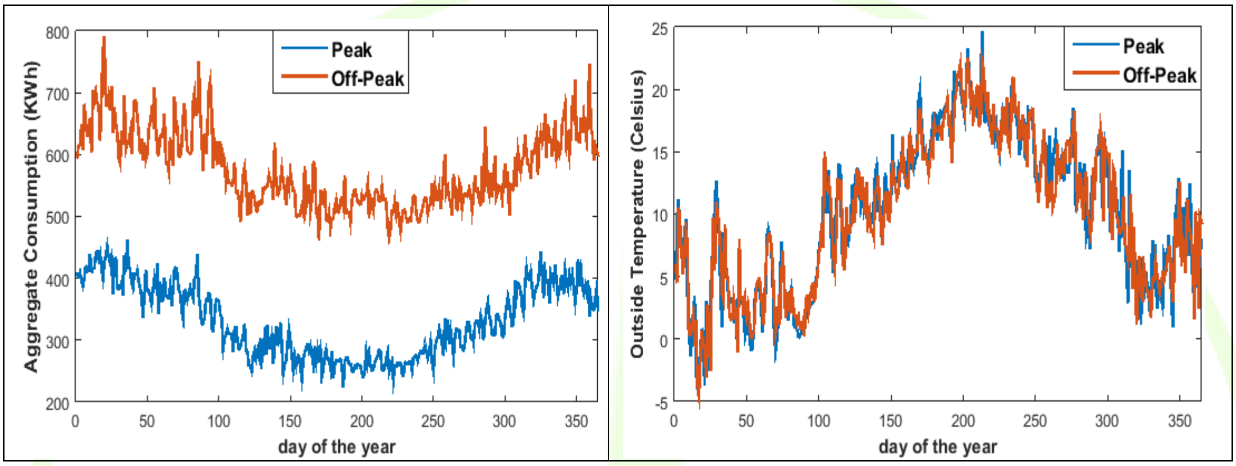

Figure 18 – The aggregate consumption (left) and the environmental temperature (right) during the peak and off-peak period for all the days of the year [16].

6.3.4 Regression and model definition

Figure 18 is very illustrative for justifying also the approach for the subset of data given as input to the regression process, targeting to provide the consumption profile of each user, i.e., for estimating the user-specific coefficients of the demand function for all the three models. More specifically, for each day of the year that a prediction of consumption is derived (see in the next section for a concrete definition of this subset), the "training period" of the regression process is considered as the last thirty days before the one with prediction interest. For instance, for the estimation of the consumption at the 31st day of January, the data-subset consisting by the records in the interval between the 1st and the 30th day of January (including the edges) was provided as input to the models. The next training period is from 2nd of January to 31st of January for a prediction on the 1st of February and so on. As Figure 18 depicts, this choice is justified by the fact that the fluctuations of both the temperature and the consumption are narrow within each of these intervals and consequently it is reasonably anticipated that the training data closely reflects the consumption under prediction. From the above description, it becomes apparent that the coefficients are not just "user" but also day-specific since the training period of the models is constantly sliding. For clarity reasons it is mentioned that the evaluation process of the proposed models does not include the first 30 days of the year (since they are necessary as the first training period), the days from 200-270 since they do not include enough dynamic pricing events and the days 270-300 (since they are necessary for the predictions of day 301).

In what follows, the three prediction models are presented, based on the assumption of whether there is load substitution between the two periods of the same day or not.

1) No substitution between periods:

This case assumes that the consumption of the user during period *j* depends only on the price and environmental temperature during this period. Consequently, the least-squares linear regression is applied on equation (25), for estimating the parameters η_j ε_j and ξ_j, resulting to the "**Simple**" prediction model.

$$Q_{j,t}^{pred} = e^{\eta_j+\varepsilon_j lnP_{j,t}+\xi_i T_{j,t}}, j\epsilon\{p, op\} \tag{26}$$

2) Substitution between periods:

This case assumes that the consumption of the user during the period j depends on the prices and the environmental temperatures of both periods within the day. Thus, the least-squares linear regression is applied on equation (19) for estimating the coefficients $\mu\pi_{p,op,t}, \varepsilon_{sub}$ and β:

The problem with equation (19) is the fact that it provides the ratio of the consumptions during the two periods, but not their values. Targeting to deal with this problem, two approaches are followed which lead to the corresponding prediction models:

The **Normalized** model estimates the consumption of each period as a function with respect to the consumption on the other. More particularly:

$$Q_{p,t}^{pred} = Q_{op,t} e^{\mu\pi_{p,op,t}+\varepsilon_{sub} ln\left(\frac{P_{op,t}}{P_{p,t}}\right)+\beta(T_{p,,t}-T_{op,t})} \tag{27}$$

And

$$Q_{op,t}^{pred} = \frac{Q_{p,t}^{pred}}{e^{\mu\pi_{p,op,t}+\varepsilon_{sub} ln\left(\frac{P_{op,t}}{P_{p,t}}\right)+\beta(T_{p,t}-T_{op,t})}} \tag{28}$$

Notice that in the former equation which estimates the peak-consumption, the off-peak consumption $Q_{op,t}$ is known (since this period proceeds), thus the actual value from the dataset is used. In the latter case, $Q_{p,t}^{pred}$ is derived from the "Simple" model.

The **Enriched** model applies the estimated coefficient $\mu\pi_{p,op,t}, \varepsilon_{sub}$ and β to the equations (23) and (24) for its predictions. The only unknown parameter in these equations is the budget m_t. In what follows, an analytical description of the approach for its estimation is presented.

Let $TR = \{t-30, t-29, \dots, t-1\}$ be the set of days consisting the training period, provided as input to the model for predicting the consumption at day t. Then $UTR = \{s \in Tr | P_{p,s} \neq P_{op,s}\}$ is the set of days when the prices of the two periods are not equal, i.e., the days when the load substitution may occur. The budget is estimated as the average budget spent by the user during these days. In mathematical terms:

$$m_t^{pred} = \frac{1}{|UTR|} \sum_{t \in UTR} P_{op,t} Q_{op,t} + P_{p,t} Q_{p,t} \tag{29}$$

For clarity reasons, it is mentioned that the "Simple" model may be applied during any day, independently if there is a difference between the prices announced at the two periods, while the other two models are applied only during the days when the aforementioned difference is realized, i.e., during these days when consumption shifting between the two periods is expected to occur.

6.3.5 Accuracy of the proposed models

This section evaluates the proposed models with respect to their accuracy for the prediction of the individual (for each user) and the aggregate (of all users) consumption. The evaluation process mainly focuses on the days when the price during the peak period is higher compared to the one applied during the off-peak, i.e., at these days when the load shifting is expected to occur (the effect that both the Enriched and the Normalized models are designed to capture). For the opposite case, i.e., when the off-peak price is higher compared to the peak-one, the findings are similar (in terms of the models' performance ranking) and consequently their presentation is omitted for simplicity reasons. Before proceeding, it is clarified that throughout this section the models are not user-specific, meaning that all the users are characterized by the same demand function and are differentiated by means of its coefficients. This process results to useful conclusions for the performance of each model (separately), which guide the design of hybrid models in the next section.

6.3.5.1 Cross performance evaluation

The basic evaluation metrics to be utilized are the "Mean Absolute Percentage Error" (MAPE) and the "Absolute Percentage Error" (APE) for the individual and the aggregate consumption respectively. In what follows, their concrete definition is provided.

Let $D = \{31,32, \dots 200, 300, 301, \dots, 365\}$ be the set of days that the models provide prediction results, then $H_p = \{t\epsilon D | P_{p,t} > P_{op,t}\}$ is the set which includes the days with evaluation interest. The mathematic formulation of the MAPE for the prediction of individual consumption of each user i during period j is as follows, where $|H_p|$ stands for the cardinality of the set H_p.

$$MAPE_{i,j}^{H_p} = \frac{1}{|H_p|} \sum_{t\epsilon H_p} \frac{|Q_{i,j,t} - Q_{i,j,t}^{pred}|}{Q_{i,j,t}}, j \in \{p, op\} \tag{30}$$

Let $Q_{j,t} = \sum_{i=1}^{M} Q_{i,j,t}$ be the aggregate consumption of all users during period j at day t, where M stands for the total number of end-users (163 in this experimental setup). Then, the mathematical formulation of the APE for the prediction of the aggregate consumption during period j for each day $t\epsilon H_p$ reads as follows:

$$APE_{j,t}^{H_p} = \frac{|Q_{j,t} - Q_{j,t}^{pred}|}{Q_{j,t}}, \forall\ t\epsilon H_p, j \in \{p, op\} \tag{31}$$

Figure 19 (left), depicts the $MAPE_{i,op}^{H_p}$ of all the user for all the proposed prediction models, plus for an additional theoretical artifact (named "Enriched - Actual Budget" and "Enriched-AB" its abbreviation), which differs with the Enriched by the fact that the budget m_t corresponds to the actual value of this parameter. More specifically for the Enriched model the value of this parameter is estimated by Equation (29) using historical data, while for the Enriched-AB the value is calculated as $m_t = \sum_j Q_{j,t} P_{j,t}, j \in \{p, op\}$, for each day $t \in H_p$. The reason for this additional model is to demonstrate the crucial impact of the budget on the performance of the model and highlight the accurate estimation of this parameter as one of the major directions of our future work.

Concerning their comparison, the Simple model has the best performance among all, since its MAPE in average terms ($\frac{1}{M} \sum_i MAPE_{i,op}^{H_p}$) equals 27.48%. The Enriched and the Normalized models appear with almost the same performance with 39.26% and 37.28% respectively. As expected, the Enriched-AB model with 30.64% has a much better behaviour than the two former, but interestingly it is also worse than the Simple one. More specifically (see Figure 19 - right), the Simple model achieves for the 25% of the users a $MAPE_{i,op}^{H_p}$ lower than 17.7% and for half of them lower than 22.8%. The performance of the other three models in the aforementioned sequence, is 24.6% and 34%, 24.1% and 33.8%, 21.33% and 27.77% respectively.

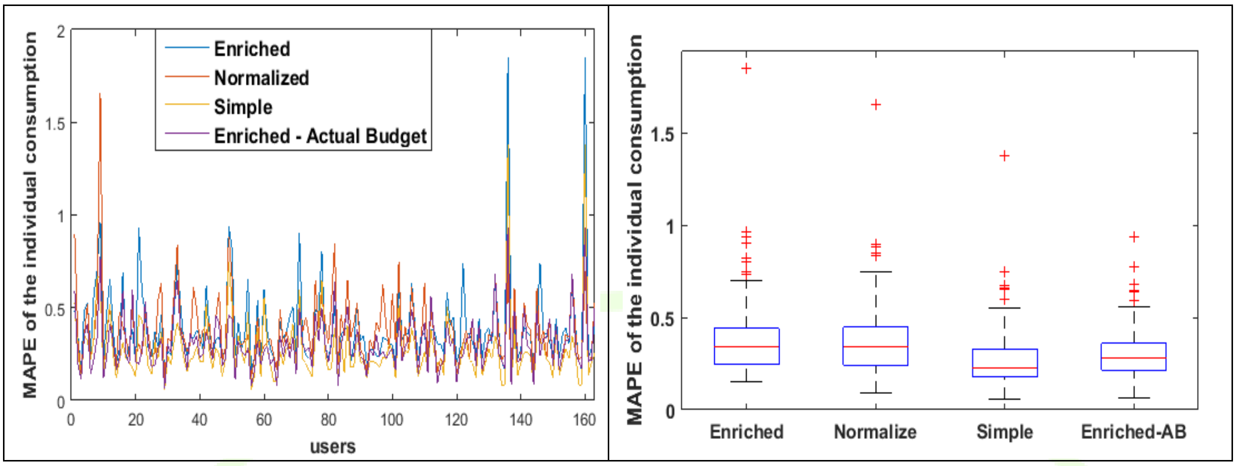

Figure 19 – MAPE of all the users' individual off-peak consumption for all the proposed models (left) and the relative boxplots (right).

In what follows it is investigated if the models that are designed to capture the users' load-shifting, perform better for those users who actually shift their consumption. The users who shift, are reasonably defined as those who appear with the following attribute: Their average off-peak consumption at the days that the peak-price is higher than the off-peak one, must be higher compared to their average off-peak consumption at the days when the prices are equal for both periods. Formally, let $E = \{t\epsilon D | P_{p,t} = P_{op,t}\}$, then the set K includes the users who shift.

$$K = \{i\epsilon M| \ \frac{1}{|H_p|} \sum_{t\epsilon H_p} Q_{i,op,t} > \frac{1}{|E|} \sum_{t\epsilon E} Q_{i,op,t}\} \tag{32}$$

According to the measurements, $|K| = 109$, meaning that 109 users out of 163 in total shift their load. The Enriched model provides more accurate results (in terms of MAPE) for only 2 users, both of whom belong in K. The Normalized model is more accurate for 22 users, 15 of whom actually shift their load. The Enriched-AB achieves closer predictions for 44 users, 29 of whom shift their load. Finally, the Simple model has more accurate predictions for 119 users, 80 of whom belong in K. Interestingly, the set composed by the 15 users whose consumption is predicted better by the Normalized model, is a pure subset of the set including the 29 users of the Enriched-AB model. This result, indicates a robust behaviour of the mechanisms designed to capture the load shifting, and is very promising for their performance when applied on a more appropriate dataset (with adequate dynamic pricing events), as the one that will be derived by the WiseGRID pilot sites.

Targeting to provide a more detailed analysis, in what follows the level of load shifting is correlated with the accuracy of each model. The metric to be used is the "Mean Percentage Load Shifting" (MPLS), defined as follows:

$$MPLS_{i,op} = \frac{\frac{1}{|H_p|} \sum_{t\epsilon H_p} Q_{i,op,t} - \frac{1}{|E|} \sum_{t\epsilon E} Q_{i,op,t}}{\frac{1}{|E|} \sum_{t\epsilon E} Q_{i,op,t}} \tag{33}$$

Notice that positive values of this metric, mean that the corresponding user actually shifts his load and vice versa. Figure 20 (left) presents the MPLS of the users who shift ($MPLS_{i,op} > 0$). The first boxplot contains all such users while the other two only those users whose MAPE is predicted more accurately by each respective model than from the Simple one ($MAPE^{x}_{i,op} < MAPE^{Simple}_{i,op}$). Notice that the Enriched-AB model has the tendency to provide more accurate predictions for those users whose load shifting is more intense, while the Normalized for those who appear with an average shifting behaviour. Figure 20 (right) presents the MPLS of the users who do not shift ($MPLS_{i,op} < 0$), with the corresponding boxplots containing the MAPE of users as described above. In this case it is more apparent that the two models provide more accurate predictions for those users whose average consumption over the days $t \in H_p$ is almost equal as the respective magnitude over the days $t \in E$, since they appear with smaller values of MPLS (even negative). These findings guide the design of hybrid prediction models (see section 6.3.6), where each individual user is associated with the appropriate demand function according to his shifting behaviour, targeting to achieve better consumption predictions both at individual and aggregate level.

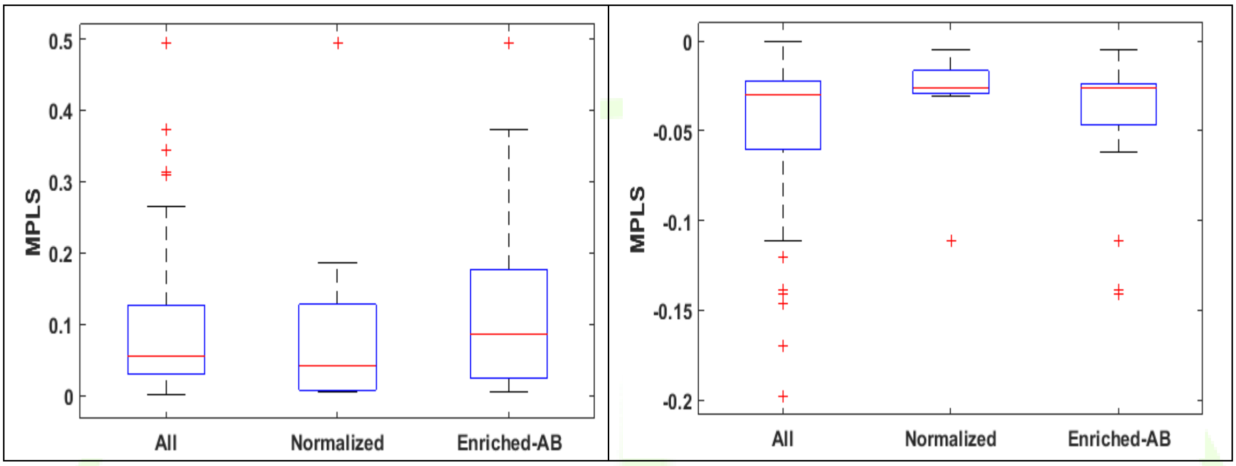

Figure 20 – Boxplots of the MPLS for those users who shift their load (left) and for those who don't (right).

In what follows, the performance evaluation of the proposed models is presented, in terms of their accuracy for the prediction of the aggregate consumption. Figure 21 depicts the $APE_{op,t}^{H_p}$ of the aggregate consumption for all the four models, and for all the days of evaluation interest: $t \epsilon H_p$. Here also, the Simple model outperforms all the others in average terms, since it appears with the lowest average value ($\frac{1}{|H_p|}\sum_{t \in H_p} APE_{op,t}^{H_p}$) which equals 4.45%. For the Enriched, the Normalized and Enriched-AB models this value equals 15.13%, 9.03% and 5.12% respectively. More specifically (see Figure 21 - right), the Simple model achieves for 25% of the days an $APE_{op,t}^{H_p}$ lower than 1.4% and for 50% of them lower than 3.3%, while the corresponding values for the other three models in the aforementioned sequence are 2.8% and 10.9%, 4.2% and 7.4%, 1.7% and 3.5% respectively.

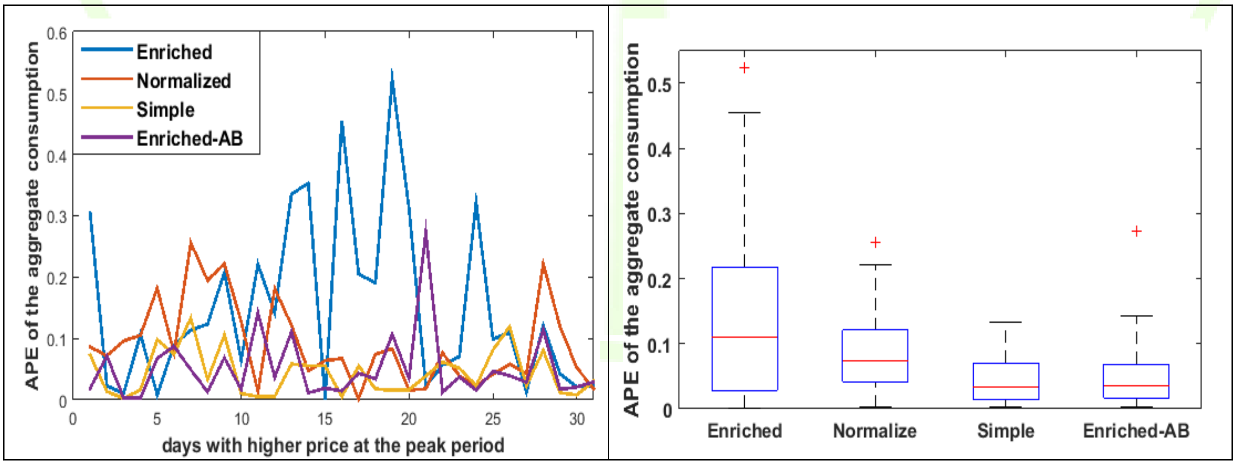

Figure 21 – APE of the users' aggregate off-peak consumption for all the proposed models (left) and the relative boxplots (right).

Despite its domination in average terms, the Simple model is outperformed for some particular days by the others. This observation is explained by the fact that the prediction of the aggregate consumption integrates the inaccuracy of all the individual predictions and consequently their over/under-estimation errors are averaged out. In what follows, a proof of this thesis is provided. Let the set $B^x = \{t \in H_p | APE_t^x < APE_t^{Simple}\}$ include the days that the model $x\epsilon\{Enriched, Normalized, Enriched - AB\}$ outperforms the Simple one and the $\mathrm{DMAPE}_{op,t} = \frac{1}{M}\sum_i \frac{|Q_{i,op,t} - Q_{i,op,t}^{pred}|}{Q_{i,op,t}}$ standing for the Daily MAPE, i.e., the Mean Absolute Percentage Error over all the individual consumptions at the specific day t.

Figure 22 (left) presents the ratio of the DMAPE of all the three models over the respective magnitude for the Simple one ($\frac{DMAPE_{op,t}^x}{DMAPE_{op,t}^{Simple}}$) for all the days $t\epsilon B^x$ ($t\epsilon B'^x$ on the right). Notice that for the Enriched and the Normalized model, the ratio is always higher than the unit (apart from once), meaning that (despite outperforming) these models provide less accurate estimates of the individual consumptions. Even though less intense, this result appears also in some cases for the Enriched-AB model, but its overall behaviour indicates more accurate predictions than the two former. On the contrary (see on the right-hand side of Figure 22), when the Simple model outperforms, the ratio is higher than the unit with only a few exceptions occurring at the days when the performance (w.r.t. the APE) of the two respective models is relatively close. Still, even in this case the ratio remains very close to the unit. These findings indicate that in their current from (and for this dataset) the Simple model appears with a much more robust behaviour, providing better predictions both for the individual and the aggregate consumption.

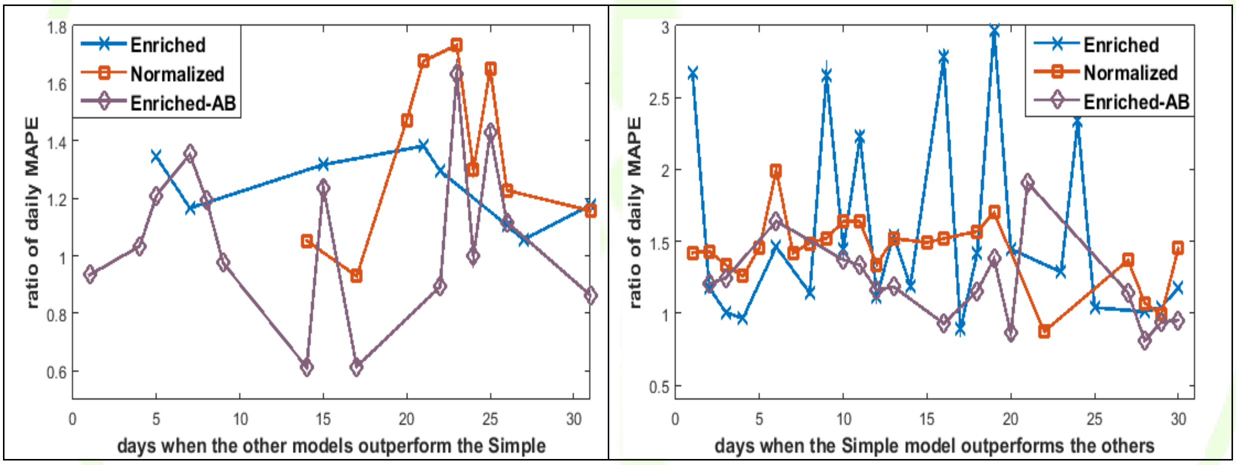

Figure 22 – Ratio of the average MAPE of all the models over the Simple one, for the days when the other models outperform the simple (left) and vice versa (right).

Targeting to investigate the comparatively poor performance of the Enriched model, Figure 23 juxtaposes the $APE_{p,t}^{H_p}$ of the aggregate consumption and the aggregate budget (for the latter parameter, the APE computation is aligned with the one for the former in Equation (31), using the corresponding predicted and the actual values of this parameter). It is easy to observe a noteworthy alignment of the two magnitudes, meaning that the accurate estimation of the budget strongly affects the performance of the model. This result is a further indication for the direction of our future work, where an extended model should take into consideration further parameters for the budget estimation (such as the income of the household). Once again it is important to emphasize that (even in its current form) we anticipate a noteworthy improvement of its performance, when it will be applied on a suitable dataset with adequate dynamic pricing events, as the one that will be derived from the WiseGRID pilot sites.

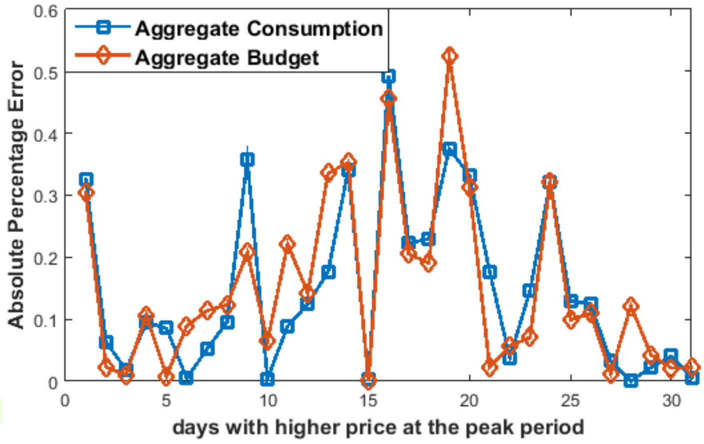

Figure 23 – Correlation between the accuracy of prediction of the aggregate budget and the aggregate consumption.

Finally, Figure 24 presents the boxplot of the $MAPE_{i,p}^{H_p}$ and the $APE_{p,t}^{H_p}$, i.e., the respective evaluation metrics for the prediction of the individual and aggregate consumption during the peak period. Quantifying the findings, here also the Simple model outperforms in average terms the Enriched and the Normalized, since they achieve an average MAPE ($\frac{1}{M}\sum_i MAPE_{i,p}^{H_p}$) equal to 40.32%, 49.96% and 41.81%. This finding is reflected to their performance in terms of the prediction of the aggregate consumption. More specifically, they achieve an average APE ($\frac{1}{|H_p|}\sum_{t\in H_p} APE_{p,t}^{H_p}$) equal to 5.72%, 15.06% and 6.55%. Interestingly, the Enriched-AB model significantly outperforms all the others with respect to both metrics, achieving 7.74% and 1.35% for the average MAPE and APE respectively.

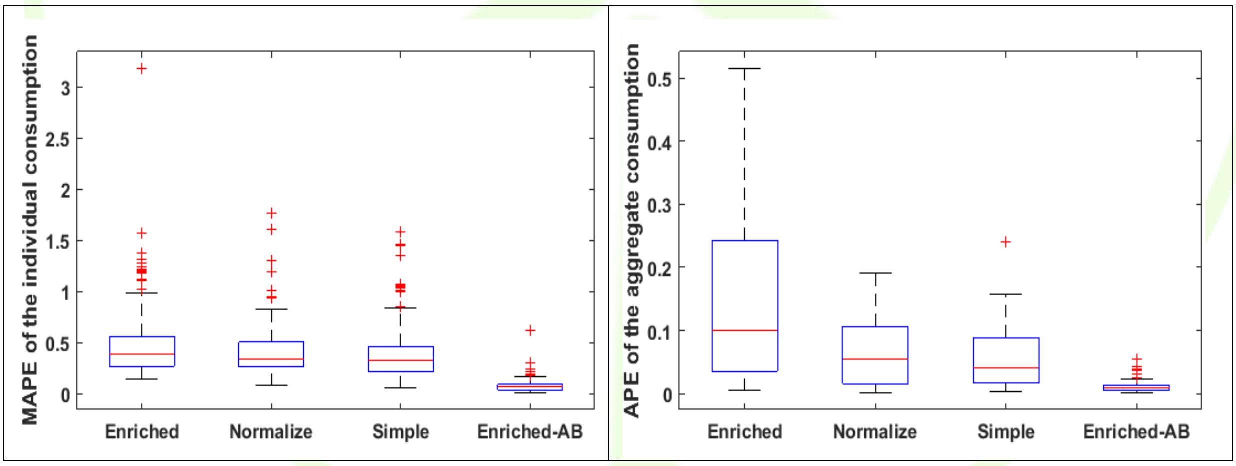

Figure 24 – Boxplots of the MAPE of the individual consumption (left) and of the APE of the aggregate consumption (right) during the peak periods for all the models.

6.3.5.2 Accuracy of the Simple model (alone)

This section presents the further evaluation of the Simple model for all the days when it provides prediction results ($t\epsilon D$), and for both the peak and the off-peak period. Notice that this part of the performance evaluation is of particular practical interest also, since a retailer is based on the predictions of this model (during the days that announces the base value of the price) for purchasing adequate energy in the wholesale market, targeting to achieve a balanced portfolio.

Figure 25 (left) presents the boxplots of the MAPE of the users' individual consumption, for the peak and the off-peak periods. Their average values ($\frac{1}{M}\sum_i MAPE^D_{i,p}$, $\frac{1}{M}\sum_i MAPE^D_{i,op}$) are 37.02% and 25.49% respectively. Similarly, Figure 25 (right) depicts the APE of the aggregate consumption for both periods, and their average values ($\frac{1}{|D|}APE^D_{p,t}$, $\frac{1}{|D|}APE^D_{op,t}$) are 6.64% and 4.93%. These findings reveal that the model maintains almost the same performance over this extended evaluation set, indicating its robust behavior.

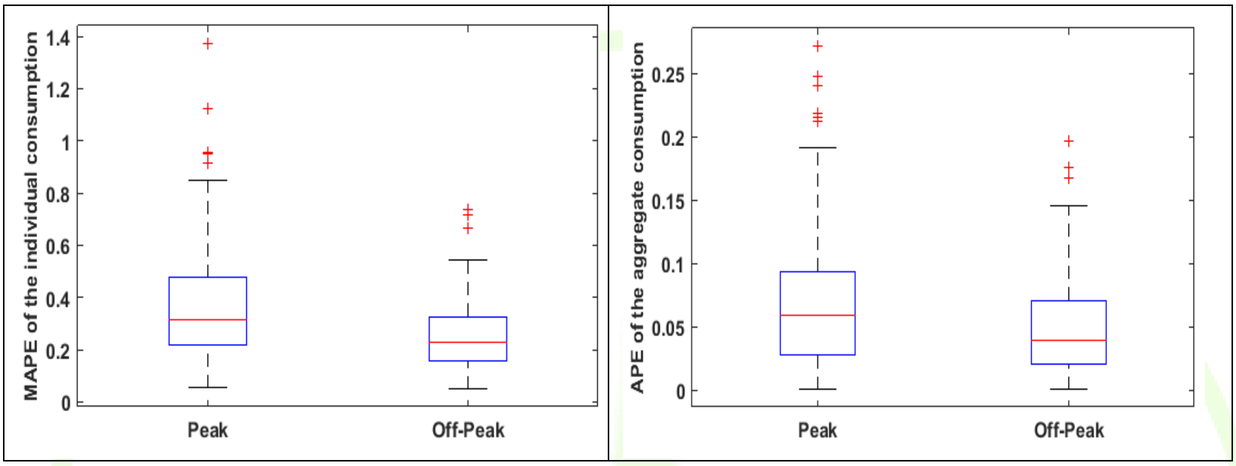

Figure 25 – Accuracy of the predictions provided by the Simple model over all the days of the year. Boxplots of the MAPE of the individual consumption (left) and of the APE of the aggregate consumption (right) during both the peak and the off-peak periods.

6.3.6 Hybrid model: Design and accuracy

The previous section followed the "one model fits all" approach, in the sense that the prediction of each user's consumption relied separately to each of the proposed models. In more technical terms, the demand functions were not user-specific, but the users were differentiated only with respect to their coefficients. This approach did not fully utilize the design properties of the proposed models. For instance (see Figure 20-left and its explanation), the Enriched-AB model appears to provide more accurate predictions for the consumption of the users who have an intense load-shifting behaviour. Thus, the main idea in this section is to identify those users who are characterized by this attribute and associate the prediction of their consumption with the models that are designed to capture this effect, while keep using the Simple model for the rest of the users. This approach results to three hybrid models, namely $H.x$ where $x\epsilon\{Enriched, Normalized, Enriched - AB\}$, which arise from the combination of the Simple model with each one of models in the set x.

In what follows, a concrete description of the criterion which separates the users according to their shifting behaviour is presented. Let $TR = \{t-30, t-29, \dots, t-1\}$ be the set of days consisting the training period (common for all the models), provided as input to each model for predicting the consumption at day t. Then $HTR = \{s \in Tr | P_{p,s} > P_{op,s}\}$ is the set of days within the training period when the peak-price is higher compared to the off-peak and $ETR = \{s \in TR | P_{p,s} = P_{op,s}\}$ is the set of days when the price is equal for both periods. The parameter $PLS_{i,t}$ measures the "Percentage Load Shifting" that the user i performed during the training period before day t.

$$PLS_{i,t} = \frac{\frac{1}{|HTR|}\sum_{t\epsilon HTr} Q_{i,op,t} - \frac{1}{|ETR|}\sum_{t\epsilon ETr} Q_{i,op,t}}{\frac{1}{|ETR|}\sum_{t\epsilon ETr} Q_{i,op,t}} \quad (34)$$

The set KTR_t includes the users with a positive value of PLS ($PLS_{i,t} > 0$), i.e., those users who performed load-shifting in the recent past before day t.

$$KTR_t = \{i = 1,2, \dots , M|PLS_{i,t} > 0\} \tag{35}$$

Finally, the criterion which separates the users according to their shifting behaviour is as follows

$$c_{i,t} = PLS_{i,t} - \frac{1}{|KTR_t|} \sum_{i \epsilon KTR_t} PLS_{i,t} \tag{36}$$

If $c_{i,t} > 0$, meaning that the data of the recent past reveal that the user i performs more intense load-shifting than the average, then the prediction of his consumption by the model $H.x$ is the same as the one provided by the model x. On the contrary, if $c_{i,t} < 0$, , then the Simple model provides the prediction.

Figure 26 (right) presents for each day of evaluation interest ($t \epsilon H^p$), the number of users who were found to satisfy the aforementioned criterion. Notice that their number does not depend on the prediction model, since the characterizing criterion is based only on actual data derived by the dataset. Figure 26 (left) presents the $APE_{op,t}^{H_p}$ of all the models for the days $t \epsilon H_p$ and the interest mainly focuses to the hybrid models, since for the Simple one the performance is identical as in Figure 21. In average terms ($\frac{1}{|H_p|} \sum_{t \in H_p} APE_{op,t}^{H_p}$) the Simple model achieves 4.45% (as in section 6.3.5.1), while the H.Enriched and the H.Normalized improve their performance and achieve 5.22%, and 7.99% respectively. Interestingly, the H.Enriched-AB model achieves 4.03%, meaning that it does not only improve its performance but also outperforms (even marginally) the Simple one. This latter finding indicates that the approach with the user-specific demand functions is meaningful, since it predicts the consumption with higher accuracy.

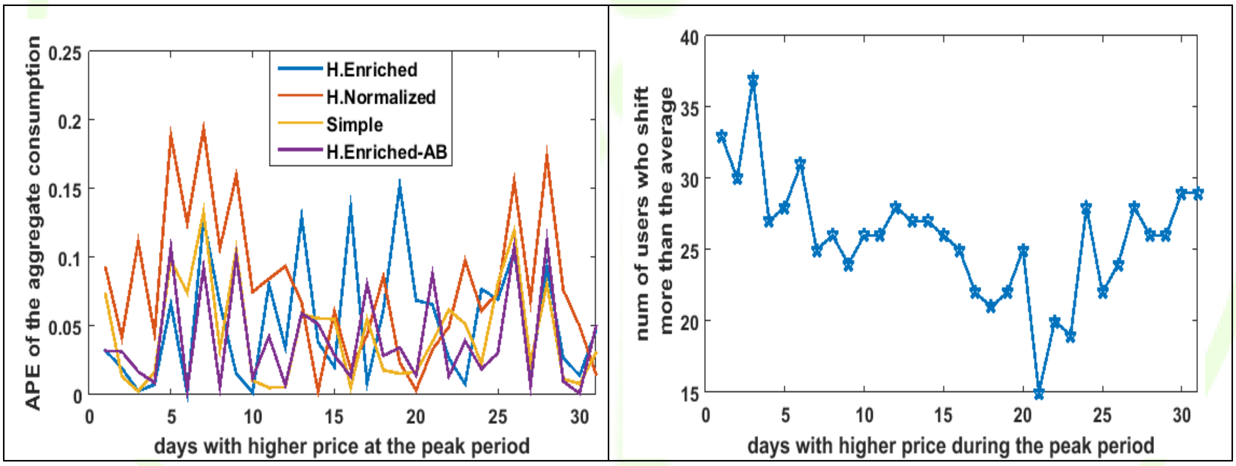

Figure 26 – APE of the aggregate consumption for all the three models (left) and the users who satisfy the criterion (right).

Concerning the performance of the models with respect to the individual consumption of each user, they achieve an average MAPE ($\frac{1}{M} \sum_i MAPE_{i,op}^{H_p}$) equal with 27.48%, 29.76%, 28.95% and 27.30% in the aforementioned sequence. This result indicates that the better performance of the H.Enriched-AB model is due to its better predictions of each user's consumption and does not rely on counterbalances of the individual misestimations when summed up.

For completeness reasons, it is mentioned that the performance of additional hybrid models was also investigated, arising by the application of the criterion which separates the users between those who shift and those who don't ($PLS_{i,t} > 0$ and $PLS_{i,t} < 0$ respectively). In this case also, the H.Enriched-AB model outperforms in average terms the Simple one, achieving an average APE of 4.19%. This is a further indicator that the aforementioned approach is meaningful, and is expected that the hybrid models will provide even more accurate predictions (compared to the Simple one) when applied at a dataset providing more dynamic pricing events.

6.3.7 Computation of the dynamic price

This section introduces an algorithm to be utilized by the retailer in the case that he aims to implement an implicit DR campaign. More specifically, the algorithm computes the value of the dynamic price to be applied, targeting to achieve a collective reaction of all the users such that their aggregate consumption results to a specific level of load curtailment. The achieved curtailment is quantified by means of comparison with the baseline load of the users, i.e. their average load observed during the days of the recent past, when the base-value of the price was announced by the retailer (in the absence of DR events – see Equation (37) for a concrete definition).

The algorithm utilizes the demand function with its user-specific coefficients derived from the regression process, (which predicts the consumption of each individual user with respect to the price and the temperature) and dynamically updates the level of the price until the aggregated predicted consumption reaches the desired level. For clarity reasons, it is mentioned that the aforementioned iterative process for the identification of the suitable value of the price is virtually executed before its actual application, i.e. the retailer announces to its clientele only the value provided by the algorithm.

In what follows, the pseudocode of the algorithm is presented. The considered scenario assumes that the retailer aims to achieve a specific reduction in KWh (notated as ΔQ_{target}), compared to the aggregate baseline load of its clientele (notated as Q^{base}).

Algorithm: Identification of the dynamic price for the implementation of implicit DR

$set\ p = p_b$ %The identification of the dynamic price is initiated with the value of the base price.

$set\ update\_step$ %The update step of the price, with a small positive value: $0 < update\_step \ll 1$.

$set\ \ Q^{base} = \sum_i Q_i^{base}$ %Computation of the aggregate baseline load of all the end-users

$set\ \Delta Q_{target}$ %Set the desired decrease: Input for the algorithm, provided by the retailer.

$set\ \Delta Q = Q^{base} - \sum_i Q_i^{pred}(p,T)$ %Computation of the curtailment with respect to the initial price value and the temperature.

$while\ \Delta Q < \Delta Q_{target}$ % Iterative process: Check if the achieved reduction meets the desired one.

    $set\ p = p + update\_step$ % Increase the price, targeting to achieve additional reduction.

    $set\ \Delta Q = Q^{base} - \sum_i Q_i^{pred}(p,T)$ % Computation of the curtailment w.r.t. the updated price. value and the temperature.

$end\ while$

$set\ p_d = p$ %The output of the algorithm: the price to be applied during the DR event

In what follows, the outcome of the algorithm is presented, for all the days when the peak price is higher than the base one and the aggregate consumption of the users is lower compared to their baseline load (dataset). For simplicity reasons, only the predictions of the individual consumptions that are provided by the Simple model have been considered.

Before proceeding, the concrete computation of the aggregate baseline load is provided. Recall from the previous section that the set ETR includes those days within the training period when the price is equal during the peak and the off-peak period. The considered baseline consumption of user i for the peak period of day t is defined as follows:

$$Q_{i,p,t}^{base} = \frac{1}{|ETR|} \sum_{t \in ETR} Q_{i,p,t} \tag{37}$$

Additionally, let $Q_{p,t}^{base} = \sum_i Q_{i,p,t}^{base}$ be the considered aggregate baseline load at day t and recall that the $Q_{p,t} = \sum_{i=1}^{M} Q_{i,p,t}$ is the actual aggregate consumption at day t. In the following experiments, the desired curtailment is set equal to the actually observed consumption decrease, as computed by the provided records (dataset).

$$\Delta Q_{target} = Q_{p,t}^{base} - Q_{p,t} \tag{38}$$

Figure 27 (left), depicts the output of the algorithm, i.e., the value of the identified dynamic price. The aforementioned values are juxtaposed with the prices actually applied during these days, emphasizing that the target here is not a performance evaluation of the algorithm by means of their between comparison, since the actual objective of the retailer when the dynamic prices were applied is unknown (for instance the retailer may followed a different approach for the computation of the baseline load, than the one described above). Thus, the target here is limited just to demonstrate that the algorithm concludes to specific values and that they lie within a reasonable interval around the actual values from the dataset, (as it actually happens apart from two cases when the identified price is relatively much higher compared to the applied one). Finally, Figure 27 (right) depicts the reduction of the aggregate consumption compared to the baseline one, for those days when the dynamic price was applied.

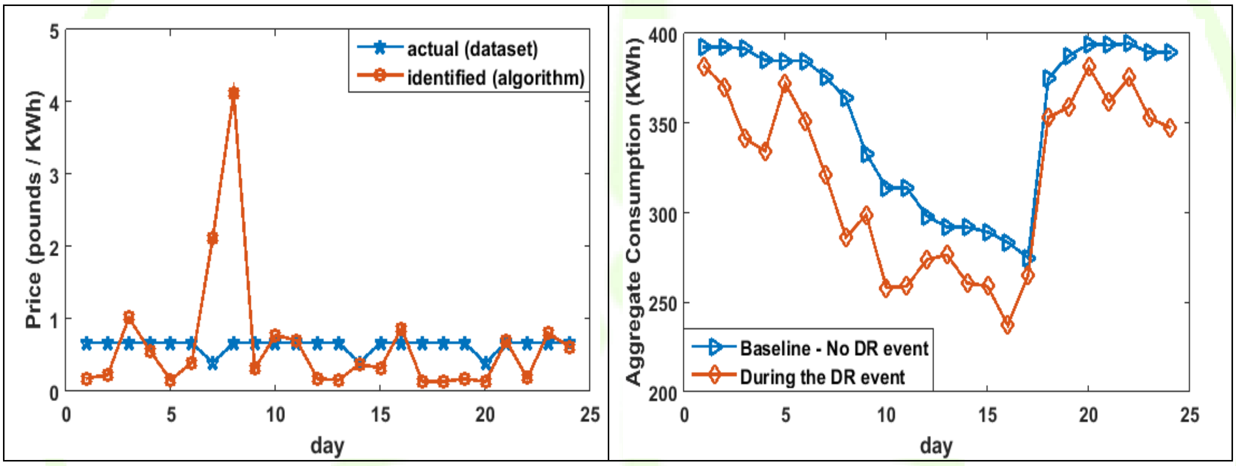

Figure 27 – The values of the dynamic price identified by the algorithm and provided by the dataset (left) and their impact on the peak aggregate consumption (right), for all the days of the year when a dynamic peak price was applied and the actual consumption was lower than the baseline one .

6.4 ELECTRIC VEHICLE DEMAND FLEXIBILITY MODEL

6.4.1 Brief description

The main objective of the Electric Vehicle Flexibility Algorithm is being able to provide the possible variations over the initially foreseen charge profile for a certain electric vehicle through the day. This definition implies that the flexibility algorithm must have in advance a time and demand forecast information for every location where the electric vehicle will be plugged on. The output provides the charge and discharge options for an electric vehicle specifically, i.e., flexibility within the V2G frame.

The operation of the algorithm is based on a minimum State of Charge required by the end of the period. This state of charge is the minimum required for the user's needs; for this reason, this end-period state of charge could be higher than the expected one but never lower. This feature provides the possibility of a flexibility algorithm.

If there are not any requirements, the algorithm behaviour is the same than the current charge points. An example of this profile is described in Figure 28.

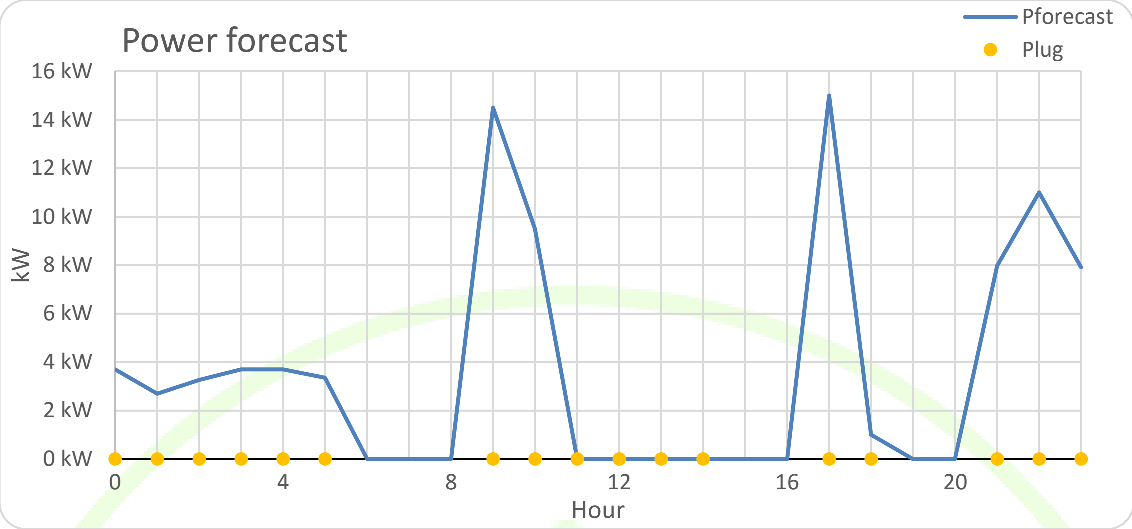

Figure 28 – EV initial situation profile

With this first approach, the forecast information must provide data about the initial state of charge and minimum state of charge at the end of the period. On the other hand, it must provide technical information, such as maximum power for charging or discharging. With this information, the plugged time is divided into periods for facilitating the calculus process.

This information is enough to provide a flexibility output. However, this output is not a static value, since every new setpoint could change the flexibility for the subsequent ones. An example of flexibility estimation is shown in Figure 29.

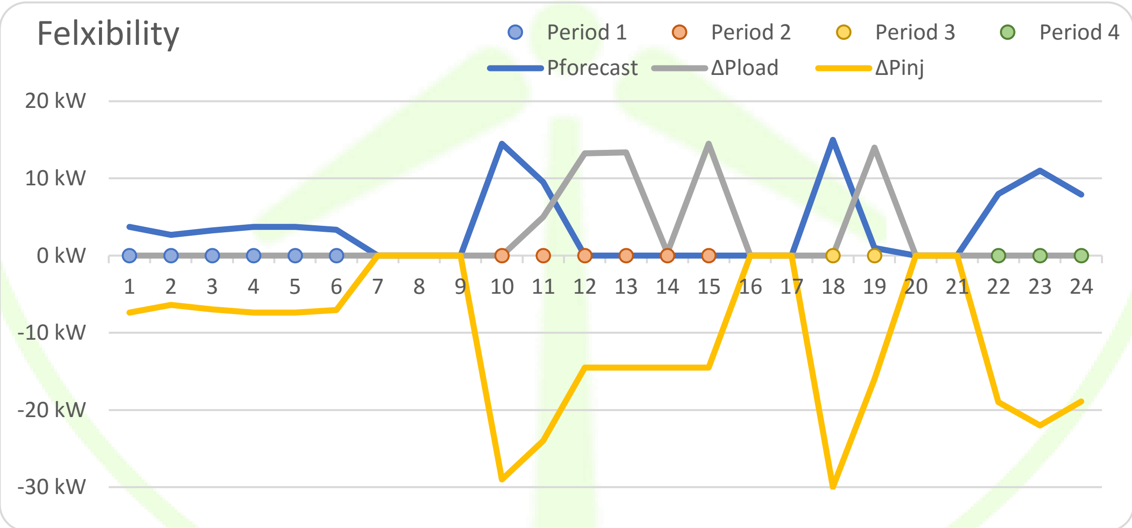

Figure 29 – Flexibility estimation graph

In order to improve the comprehension of the algorithm results, the State of Charge evolution is exposed in Figure 30. A study of this graph allows to understand the algorithm behaviour when the battery of the Electric Vehicle is saturated (i.e., SOC reaches 100% capacity), adding a restriction to the flexibility output.

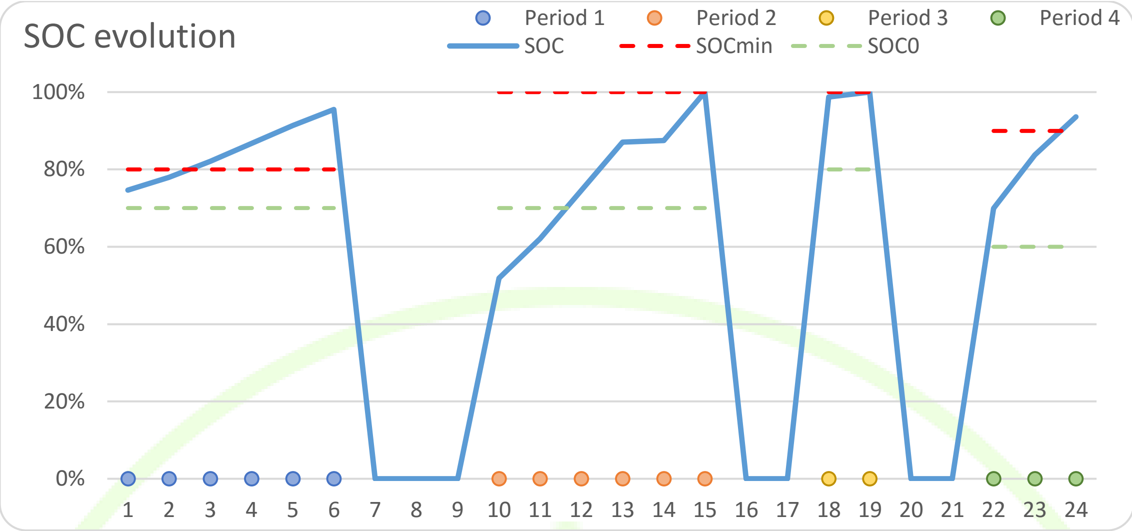

Figure 30 – Electric Vehicle State of Charge evolution

This algorithm is scalable and allows to aggregate several Electric Vehicles for an Aggregated Flexibility.

6.4.2 Baseline energy asset model

In the following paragraphs the main variables involved in Electric Vehicle Demand Flexibility Estimation will be described. First of all, it is required to have information about certain needed parameters for characterizing the model:

- Parameters
  - P_{charge}: Maximum charge power provided by the charger. It depends on the connection mode and the charge point (kW).
  - $P_{discharge}$: Maximum injection power provided by charger. It depends on connection mode and the charge point (kW).
  - P_{con}: Maximum power available for the electrical installation where the charger is located (kW).
  - t: Timestep corresponding to the input arrays (h).
  - SOC_{MAX}: Maximum State of Charge of the Electric Vehicle (kWh)
- Requirements
  - P_{req}: Charger power setpoint. It is the control variable of the process. Its value will be achieved by the charge point if the flexibility possibilities allow this setpoint (kW), either for charging (positive sign) or for discharging (negative sign).

On the other hand, for being able to provide flexibility, a forecast input is needed for the following variables:

- Forecast
  - P_{load}: Expected demand of the place (e.g.: user's house, workplace…). It can be irrelevant at certain places (e.g.: public charge stations), in these cases, the expected demand is equal to zero and the only restrictions come from the charge point characteristics (maximum load P_{charge} and maximum injection power $P_{discharge}$ allowed) (kW).
  - SOC_0: Initial State of Charge for certain period (kWh).
  - SOC_{min}: Minimal State of Charge at the end of the period (kWh).
  - Period: Time when EV is connected to charger point, from the last unplugged time to the next unplugged time (-).

The Electric Vehicle Flexibility Algorithm outputs are the variables which represent the flexibility. These outputs will be expressed as arrays corresponding to the input ones:

- Flexibility

- SOC: Expected State of Charge of the Electric Vehicle (kWh).
- $P_{forecast}$: Expected Power consumption from the grid (kW).
- ΔP_{load}: Available power increment over the $P_{forecast}$ (kW).
- $\Delta \mathrm{P}_{\mathrm{inj}}$: Available power decrement over the $P_{forecast}$ (kW).

6.4.3 Model definition

Each calculation step corresponds to a variable that could be one of the flexibility variables or an internal variable of the algorithm. Therefore, they will be expressed as a definition formula followed by the corresponding restrictions (blue equations) that apply to each variable. Before presenting the equations, the following advices must be considered:

- The "t" sub index means that the variable is calculated for every t-time step.
- The "p" sub index means that the variable depends on the p plugged period.
- Every calculus must be done with the corresponding period parameters (charge point characteristics, maximum available power from the grid, expected demand, plugged time and initial and minimum SOC).

The following equations define the Electric Vehicle Flexibility Algorithm:

- $P_{chr_{av-t}}$: Charge available power after considering the other consumptions.

$$P_{chr_{av-t}} = P_{con_t} - P_{load_t}$$
$$0 \leq P_{chr_{av-t}} \leq P_{chr_{VE-t}} \tag{39}$$

- $P_{disch_{av-t}}$: Charge available power.

$$P_{disch_{av-t}} = Pdisch_{VE_t} \tag{40}$$

- $P_{req_{fil-t}}$: Required power limited to available charge and discharge power.

$$P_{req_{fil-t}} = P_{req_t}$$
$$P_{disch_{av-t}} \leq P_{req_t} \leq P_{chr_{av-t}} \tag{41}$$

- P_{p_t}: Remaining power to achieve minimum State of Charge (for a period), considering the stored energy, the current power and the future power flow without requirements (maximum charge power).

$$P_{p_t} = SOC_{\mathrm{min}_t} - \left(SOC_{t-1} + P_{req/chr_{-t}} \cdot \Delta t \sum_{t+1}^{t_p} \vec{P}_{chr_{av}} \cdot \Delta t \right) \tag{42}$$

- P_{eff_t}: Power flow between Electric Vehicle and the grid.

$$P_{eff_t} = \begin{cases} \dfrac{SOC_{\mathrm{min}_t} - \left(SOC_{t-1} + \sum_{t+1}^{t_p} \vec{P}_{chr_{av}} \cdot \Delta t \right)}{\Delta t}, & P_{p_t} \cdot \Delta t < 0 \\ P_{req/chr}, & P_{p_t} \cdot \Delta t \geq 0 \end{cases} \tag{43}$$
$$P_{max_{inj-t}} \leq P_{eff_t} \leq P_{\mathrm{max}_{chr-t}}$$

- $P_{req/chr}$: Power required for calculus.

$$P_{req/chr} = \begin{cases} P_{req_{fil-t}}, & P_{req_{fil-t}} \neq 0 \\ P_{max_{load-t}}, & P_{req_{fil-t}} = 0 \end{cases}$$
$$P_{max_{inj-t}} \leq P_{req/chr} \leq P_{max_{chr-t}} \tag{44}$$

- $P_{max_{inj-t}}$: Maximum injection available power, considering SOC and limits, for achieving, at least, the minimum SOC.

$$P_{max_{inj-t}} = \frac{SOC_{\min_p} - \left[SOC_{t-1} + \sum_{t+1}^{t_p} \vec{P}_{chr_{av}} \cdot \Delta t\right]_0^{SOC_{MAX}}}{\Delta t} \tag{45}$$

$$P_{disch_{av-t}} \leq P_{max_{inj-t}} \leq 0$$

- $P_{max_{chr-t}}$: Maximum charge available power, considering SOC and limits, for achieving, at least, the minimum SOC.

$$P_{max_{chr-t}} = \frac{\left[SOC_{t-1} + P_{chr_{av}-t} \cdot \Delta t\right]_0^{SOC_{MAX}}}{\Delta t} \tag{46}$$

$$0 \leq P_{max_{chr-t}} \leq P_{chr_{av-t}}$$

- SOC_{max_t}: Maximum SOC available for charging.

$$SOC_{max_t} = SOC_{MAX} - \sum_{t+1}^{t_p} \vec{P}_{req_{fil}} \Delta t \tag{47}$$

$$0 \leq SOC_{max_t} \leq SOC_{MAX}$$

$$SOC_{t-1} - P_{max_{inj-t}} \leq SOC_{max_t} \leq SOC_{t-1} + P_{max_{chr-t}}$$

With these variables, it is possible to obtain the flexibility variables as follows:

$$SOC_t = SOC_{t-1} + P_{eff_t} \tag{48}$$

$$0 \leq SOC_t \leq SOC_{max_t}$$

$$P_{forecast_t} = \frac{SOC_t - SOC_{t-1}}{\Delta t} \tag{49}$$

$$P_{disch_{av-t}} \leq P_{forecast_t} \leq P_{chr_{av-t}}$$

$$\Delta \mathrm{P}_{\mathrm{load}_t} = P_{chr_{av-t}} - P_{forecast_t} \tag{50}$$

$$\Delta P_{load_t} \geq 0$$

$$\Delta \mathrm{P}_{inj_t} = P_{disch_{av-t}} - P_{forecast_t} \tag{51}$$

$$\Delta P_{inj_t} \leq 0$$

6.4.4 Model estimation

The calculus process is summarized in Figure 31, where $Charger$ refers to any information about the charger station, and SOC_{-1} refers to the State of Charge of the previous point.

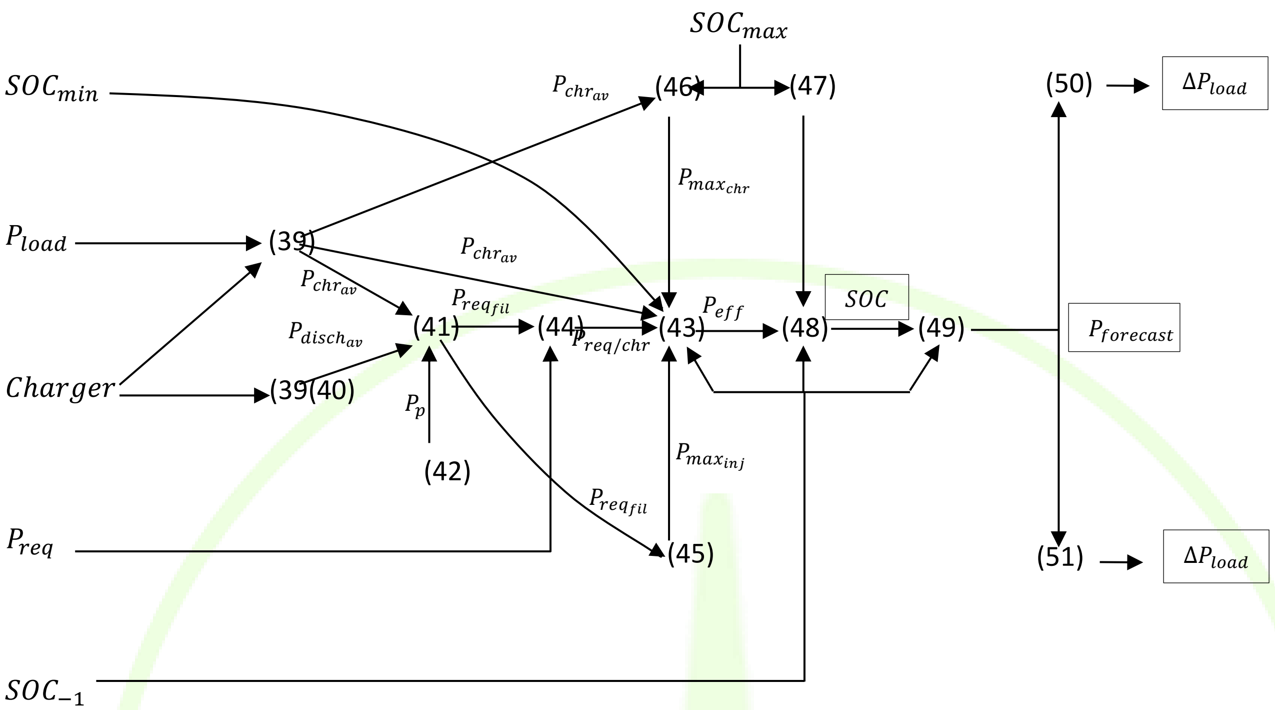

Figure 31 – Algorithm Implementation calculation scheme

7 DEMAND RESPONSE OPTIMIZATION FRAMEWORK

7.1 BRIEF DESCRIPTION

7.1.1 Purpose

In order to enable integrated and personalized energy services we need to deploy intelligent control strategies that take into account all operational aspects of the assets at hand (viz. commercial and residential buildings, batteries and EVs) while tackling grid imbalances and/or other business objectives (renewable energy sources (RES) exploitation, peak load management, etc.). These control strategies should attain to consumer behavioural patterns in order to achieve acceptance, maintain comfortable indoor conditions as well as accommodate grid services requirements.

Effective demand response strategies should continuously consolidate consumer preferences and facilitate them by providing intelligent control campaigns [21]. Such problems can be formulated as Mixed-Integer Linear/Non-Linear Programming (MILP or MINLP) optimization problems in order to eliminate grid imbalances, minimize energy demand, cost or other relevant business objectives while maintaining occupant comfort within a given set of comfort boundaries [21] or abiding to elasticity constraints of consumer demand. Solving for the objective at hand over a given time horizon while constraining consumer comfort or pricing levels within the allowed boundaries, provides a set of permissible control strategies at the building level and provides insight of available potential flexibility to actors involved (DSO, aggregator, retailer, etc.) at portfolio level.

The purpose of this chapter is to formalize and document the approach implemented for WiseGRID demand response optimization. Firstly, we describe the relevant DR strategies that are applicable to the WiseGRID context (section 7.1.2); next, we present the architecture of the DR framework (section 7.1.3) and then we provide a formalization of the optimization problem tackled in each business case (sections 7.2, 7.3)

7.1.2 Relevant DR strategies

We have to point out that two different types of DR strategies are defined in this project, covering in that way the alternative demand response business cases examined in the project; i.e. implicit and explicit demand response. The high-level description of the implicit and explicit DR strategy is presented in the following tables:

Strategy Id	Implicit Demand Response
Strategy Description	The main idea behind this functionality is to provide the Retailers a tool for managing load imbalances through novel dynamic pricing schemes. This strategy allows energy market participants to define fine-grained billing strategies for a portfolio of consumers based on their specific operational profiles (energy consumption during the night/day, energy consumption based on the day of the week etc.)
Metrics	Energy Consumption (daily load profile information) and forecasted Energy Consumption (daily load profile information), billing price levels, billing period etc.
Workflow	1. Total energy imbalance ($DQ = Q_{Production} – Q_{Demand}$) time-series for a predefined period (day-ahead) and for predefined intervals (e.g. hourly); 2. Calculation of elasticity of demand for each asset for a given set of billing prices on a time-interval basis (e.g. hourly); 3. Optimisation of portfolio elasticity based on pricing levels;

	4. One universal pricing scheme is, thereafter, broadcasted to all assets in the portfolio;

Table 9 – Implicit Demand Response Strategy

Strategy Id	Explicit Demand Response
Strategy Description	The main idea behind this functionality is to provide Aggregators a tool for responding to the demand flexibility requests in real-time through explicit demand response. This strategy allows energy market participants to cover load imbalances or DQ requests by the DSO, in the short-term future (2 hours ahead) by dispatching device-specific control requests to buildings based on their specific operational profiles (current and near-future energy consumption of devices, devices' current status, indoor ambient conditions of buildings, consumer preferences, etc.)
Metrics	Energy Consumption (daily load profile).
Workflow	1. Total energy imbalance (DQ) time-series for a predefined period (2 hours ahead) and for predefined intervals (e.g. hourly) generated based on flexibility request; 2. Calculation of demand flexibility for each device in a building participating in DR, for a set of setpoints on a time-interval basis (e.g. hourly or 15-minute intervals); 3. Filtering of buildings based on which actor is operating the tool and based on spatial restrictions; 4. Ranking of buildings based on: a. their flexibility potential, b. number of DR triggers based on historical DR data, c. DR responsiveness of each asset based on historical DR data; 5. Clustering analysis for the definition of groups of buildings with similar characteristics. Buildings are clustered in groups which reflect groups with high, medium and low ranking in a potential DR request; 6. Optimization is then performed in order to select the assets that cover an explicit DR request, based on the ranking of each asset mentioned above;

Table 10 – Explicit Demand Response Strategy

The next section presents an overview of the architecture and the various components that comprise it along with their interconnections. After the high-level view of the architecture, the following sections of this chapter respectively treat each scenario and highlight the way to deal with the demand response strategy in question.

7.1.3 Architecture overview

The DR Framework is capable of managing both explicit and implicit Demand Response campaigns, as mentioned in the previous sections. Due to the inherent differences in the functionalities required to estimate the necessary metrics (e.g. flexibility vs. elasticity), the software architecture and components that comprise the respective tools are distinct. This subsection provides a high-level view of the two architectures; the following sections will delve in more detail.

The following figure depicts an overall view of the components included in ***the implicit demand response architecture*** view. The retailer is the main actor using this view of the DR architecture in order to balance its portfolio of users. Potential day-ahead imbalances (DQ time-series) are established by production and demand forecast modules; e.g. higher production figures caused for example because of RES production, and/or higher demand, that necessitate load balancing. To this end, WiseCOOP is responsible for determining the appropriate price scheme for day-ahead application at portfolio level. This price scheme is, thereafter, broadcasted to all the interested tools (i.e. WiseCORP, WG STaaS/VPP, WiseHOME, WiseEVP).

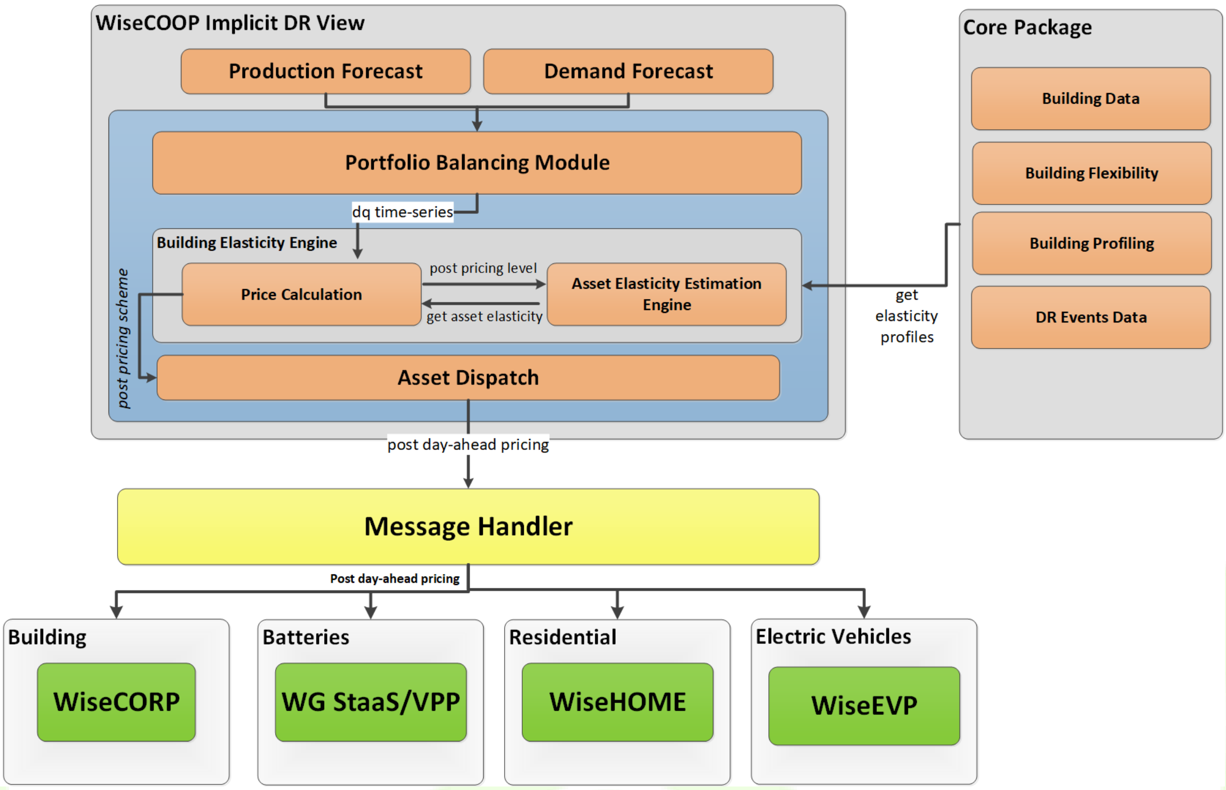

Figure 32 – Implicit Demand Response Component Architecture

A short description of the core components of the DR framework is provided:

- **Portfolio balancing module:** this module estimates the necessary demand modification that is required during the following day in order to ensure the balance of the retailer's portfolio. It uses the production (locally by retailer's generation assets) and the demand forecast in order to estimate at which moments in time and how much imbalance is expected. This information is essentially equivalent to specific requests for demand profile modification using dynamic tariff schemes.
- **Asset elasticity estimation engine:** the purpose of this component is to estimate the total portfolio demand of the retailer based on a price level it receives from the price calculation component through estimation and aggregation of the building level demand. It leverages building price elasticity models to evaluate the potential demand modification at the building level and for the entire portfolio.
- **Price calculation:** this component keeps track of the price optimization process by exploring alternative pricing levels throughout the time slots of the target day and by invoking the component above to quantify the demand alteration. The outcome is the price time-series for the target day (following day) that optimally alleviates the imbalance calculated up front.
- **Dynamic tariff dispatch:** the purpose of this component is to communicate the calculated dynamic price time-series via the WG IOP so that the other WiseGRID products receive it for their internal purposes.

Respectively, the following figure depicts the ***explicit demand response architecture***. In this case, requests are received from the network operator or other market actor who requires provision of demand flexibility; this functionality is provided through the WiseCOOP tool, which initially analyses the DR signal and consequently ranks the available assets on a multi-criteria basis. Thereafter, a selection of assets that collectively meet the needs of the DR request receive a request to provide upward or downward flexibility. Each building is then responsible to translate this flexibility request into control signals to devices through a second level of optimization.

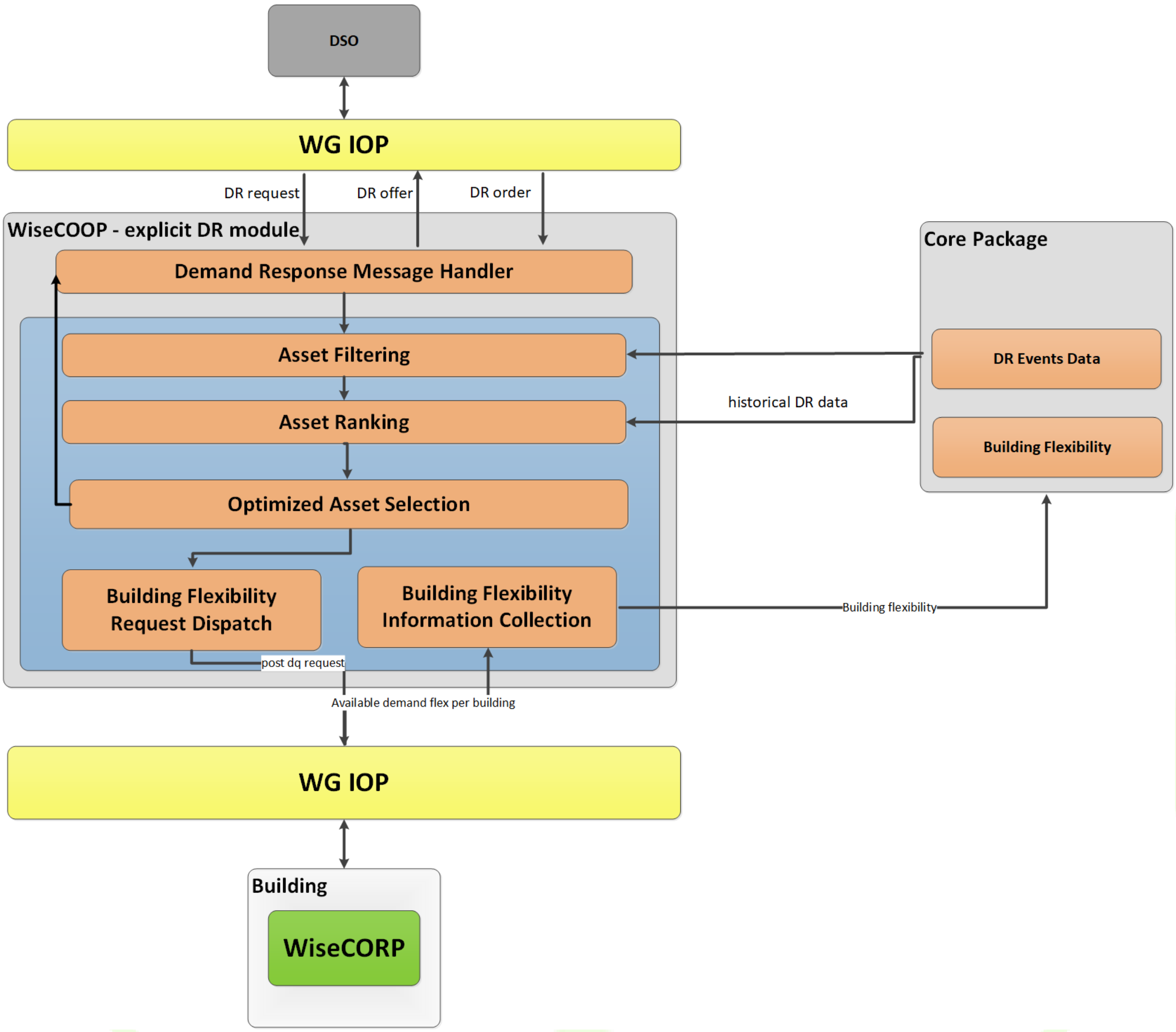

Figure 33 – Explicit Demand Response Component Architecture

A short description of the core components of the DR framework is provided:

- **Demand Response Message Handler**: this component is responsible for the interactions with the IOP in order to ensure proper information exchange. It will also perform the hand-shaking with the DSO according to the USEF specifications to facilitate the negotiations between the DSO and all the actors who are willing to provide the requested flexibility.
- **Asset Filtering**: the purpose of this module is to eliminate any asset that is not eligible to provide flexibility for the specific DR request. There may be several reasons for this, e.g. the location of its connection point on the grid, the maximum invocation number may have been reached, the asset may have declared itself unavailable due to maintenance, etc.

- **Asset Ranking**: this component aims to rank the available assets/buildings according to optimization criteria of interest to the aggregator. These may include the amount of flexibility offered, the flexibility provisions reliability of the asset, the price it requests for flex provisioning, etc.
- **Optimised Asset Selection**: the role of this component is to select the assets from the list that will comprise the asset subset that will be called to offer flexibility. Furthermore, this component will also define how much flexibility per asset should be delivered in order to optimise some objective function.
- **Building Flexibility Request Dispatch**: this component is responsible for dispatching the flexibility requests to the specific assets that have been selected to participate in the specific explicit DR campaign. It will inform the buildings about the timing and exact amount of demand modification expected to fulfil the DSO request.
- **Building Flexibility Information Collection**: this component collects the available flexibility from the various assets/buildings that have a commercial agreement with the WiseCOOP user. These flexibilities are the starting point for the explicit DR module in order perform all the aforementioned functionalities.

After giving an overview of the explicit and implicit DR architectures and briefly describing the components of the DR framework, we proceed with a more detailed description of the core components.

7.2 IMPLICIT DEMAND RESPONSE COMPONENT DESCRIPTIONS

7.2.1 Portfolio Balancing Component

The Portfolio Balancing Component receives forecasts of production and demand from the respective modules in order to calculate potential day-ahead imbalances in the portfolio of assets. This interface is at the internal RabbitMQ message handler in a publish/subscribe manner.

The following table gives an overview of the message for both production and demand forecast:

Demand forecast is requested on the AMQP queue: “forecasting_demand”
Production forecast is requested on AMQP queue: “forecasting_production”
Message properties
• reply_to: name of the queue where response will be delivered
• correlation_id: free text for query/response correlation (RPC pattern https://www.rabbitmq.com/tutorials/tutorial-six-python.html)
• Payload1:
o client_id: client identifier in the WiseGRID database.
o Horizon: number of days client wants to predict, starting from current day. From 1 to 7.
o Period: Time period between forecast samples. 15 min / 60 min. Default 60 minutes.
 	
The demand and production forecast request JSON body is as follows:
{
"Client_id": 1,
"Horizon": 1,
"Period": 60
}
 	
The JSON body for forecasting demand or production response is:

{
"errCode":0,
"forecast":
{
"1507154400":22.544,
"1507158000":21.438,
"1507161600":12.242,
"1507165200":12.116,
"1507168800":10.985,
"1507172400":12.235,
"1507176000":9.152,
"1507179600":58.837,
"1507183200":65.365,
"1507186800":22.05,
"1507190400":38.03,
"1507194000":8.861,
"1507197600":1.071,
"1507201200":15.919,
"1507204800":20.187,
"1507208400":16.721,
"1507212000":9.775,
"1507215600":2.027,
"1507219200":4.288,
"1507222800":3.249,
"1507226400":6.186,
"1507230000":2.068,
"1507233600":3.909,
"1507237200":2.478
},
"units":"kW"
}

Response properties

errCode: Error code regarding possible exceptions.

Forecast: Desired prediction formed by key value pair (“Timestamp” : Value) a) Timestamp: UNIX time seconds (UTC), b) Value: predicted value for the specified timestamp.

Units: Value units.

For each forecast to be generated, this module will take into consideration next inputs from the client:

- The client identifier. One client identifier is associated to an **aggregation** thus the same client application can request a forecast under different client identifiers.
- The period of time between two consecutive forecasting values.
- The total window horizon for the forecast output.

Subtracting demand forecast from production forecast gives the **imbalance** (DQ) time-series per defined interval. This is then fed to the Building Elasticity Engine. The latter is comprised of the Price Calculation Component and Asset Elasticity Estimation Engine that facilitate the definition of an appropriate pricing level for each interval, limiting in that way imbalances (DQ) at the portfolio level for the Retailer. In this way, a dynamic pricing scheme is produced for day-ahead based on production and demand figures at the portfolio level.

7.2.2 Price Calculation Component

We highlight the role of Building Elasticity Engine component as the fundamental component in the optimization process of the tool. The role of the price calculation component is to define pricing levels for each interval where imbalance exists. These pricing levels are then broadcasted to WiseGRID tools. The most common alternatives for dynamic pricing schemes are defined for the project:

- ***Time-of-use pricing (ToU)*** is a rate where the price per kWh depends on the time when electricity is consumed. Usually periods and prices are known well in advance, but offers where the definition of the day/night intervals may change according to the day-ahead spot price also exist. Prices can also be defined as average prices for different time periods but be directly indexed to the day-ahead spot price.
- ***Critical peak pricing (CPP)*** is a top-up rate whereby electricity prices substantially increase for the few days a year when wholesale prices are highest, but where prices are lower than average during the rest of the year. E.g. French Tempo tariff is a contract with a fixed price all year except for a maximum of 20 days with very high prices. These days are notified to customers the day before.
- With ***real-time pricing (RTP)*** wholesale electricity prices are directly passed through to final consumers and bills are calculated based on at least hourly metering of consumption, or with even higher granularity (e.g. 15 minutes). The price of such offers is composed of the wholesale price of electricity plus a supplier margin.

This component is essentially a what-if simulation engine. The algorithmic framework for this module is defined in this section.

The users will actively participate in DSM project activities, towards proactively reacting in abnormal market or grid conditions. An indicative structure of the CPP schema for a given portfolio of assets is presented in the following figure.

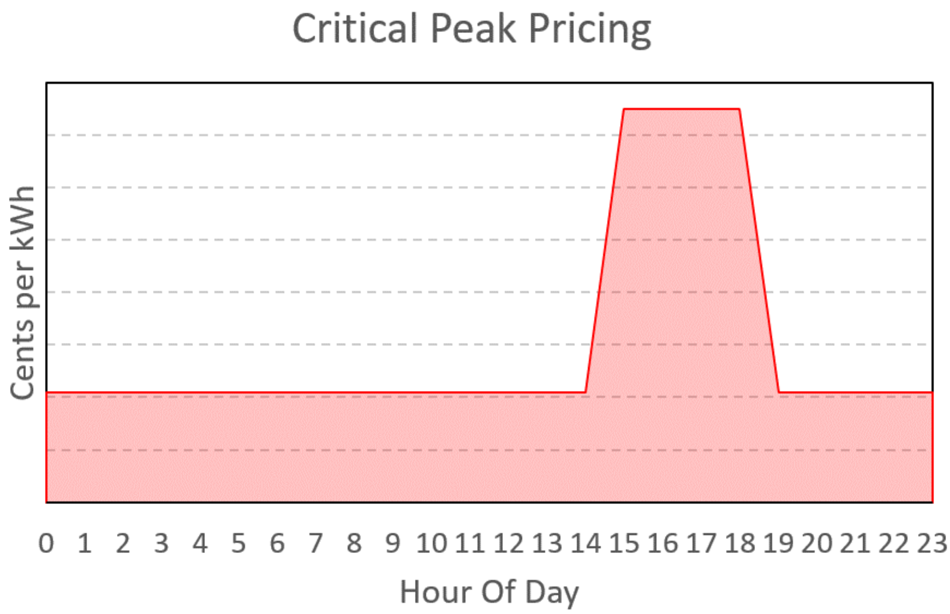

Figure 34 – Critical Peak Pricing Schema

In this case scenario, a flat tariff schema is the baseline, with a Critical Peak pricing event activated as a behavioural triggering message.

On the other hand, an indicative **ToU format** is exemplified in the following figure.

WEEKDAY			
13%	Low Tide	23:00 – 06:00	4.99p
24%	Tide Weekday	06:00 – 16:00	11.99p
19%	High Tide	16:00 – 19:00	24.99p
16%	Tide Weeknight	19:00 – 23:00	11.99p
WEEKEND			
24%	Low Tide	23:00 – 06:00	4.99p
5%	Tide Weekend	06:00 – 23:00	11.99p

Figure 35 – Time of Use Pricing Schema

The Price Calculation Component receives the DQ time-series and initiates a communication with the Asset Elasticity Estimation Engine. The optimization formulation defined below is utilized in order to define the appropriate pricing level for the given time intervals. Once the pricing levels have been defined for each interval, these are then broadcasted to the WG IOP for all the interested tools.

For implicit demand response, the elasticity modelling framework is used as described in Chapter 6 in order to predict day-ahead demand elasticity for each asset. Control variables in this case are the pricing levels defined in the contractual agreements between the interested parties. The formalization of the optimization approach used in this work is given below:

$$\begin{aligned} &\min J_k \\ &\text{s.t.} \\ &\quad u(k+j \mid k) \in [p_0, p_1, \dots, p_N] \qquad \forall\, j = 1, \dots, N_u \end{aligned} \tag{52}$$

J_k is the sum of squared residuals between flexibility requested per interval and potential flexibility at the same interval. N_u is the future control horizon; $u(k + j \mid k)$ is the control signal at time k + j, computed at time k; p_j is the discrete pricing level (implicit DR);

The objective of the implicit demand response optimization process is the **minimization of portfolio imbalances**. The objective function J_k is selected as such so that its minimization will cover any imbalances that are identified during the Portfolio Balancing step. With respect to implicit demand response at time j within the time horizon, the residual of the jth interval is defined as follows:

$$r_j = dq_j - \sum_{z=1}^{Assets} flex(u_z) \quad (53)$$

And the respective objective function to be minimized takes the following form:

$$J_k = \sum_j^{N_u} (r_j)^2 \quad (54)$$

Initially, a $DQ_{imbalance}$ is defined which is a vector of dq_j imbalances per interval (24 values, one for each hour for day-ahead); i ∈ Assets (in this case buildings in the portfolio) and u_j is the pricing signal at time j. Note that flexibility is calculated by the Asset Elasticity Simulation Engine component for a given pricing level. The selected pricing level for each interval is universally applied to all assets (buildings) as a result of the **fairness** principle. According to the assumptions of the implicit demand response business case defined in section 5.1.2, and in respect to input from the relevant actors, all customers receive and are billed according to the same energy price. Hence, all customers will receive the same retail price per interval for day-ahead.

Moreover, in respect to the elasticity of demand that each asset exhibits, we are only considering the contractually agreed pricing levels in order to balance the portfolio (implicit demand constraint).

The business objectives incorporate the implicit demand business case examined in the project, namely:

- Implicit Demand Side management (ToU, CPP & RTP) strategies to address peak load management and portfolio balancing.

The JSON representation of the internal message used for requesting a price calculation for a list of imbalances per interval is presented in the following table:

The JSON body request should have the format:


```
{
  "space": "",
  "type": "implicit",
  “dsm_request”: [
    {
       "value": "XX",
      "timePeriod":"0"
    },
    {
       "value": "XX",
      "timePeriod":"1"
    }, …,
    {
```

```
        "value": "XX",
       "timePeriod":"23"
    }
]
}
```

The configuration parameters required for the simulation analysis are: the **type of DR request** (directly associated with the type of contract); in WiseGRID project: "implicit" demand response type. The "dsm_request" is a list of objects (as defined by the Portfolio Balancing Module):

"dsm_request" in the implicit demand response case is a list of **24 objects** (day-ahead pricing) starting with timePeriod : "0", and ending to "23", and "value" indicating the imbalance identified at that particular interval.

Figure 36 – Internal Implicit DR request Interface of WiseCOOP for Retailer

For the implicit demand case, this message is used internally by the retailer in order to balance its own portfolio of assets.

In conclusion, the price calculation component is intertwined with the Asset Elasticity Estimation Engine in order to perform what-if simulation analyses. It feeds pricing levels to the Asset Elasticity Estimation Engine on request of the optimization process. In other words, once the price calculation component processes imbalance data, the outcome is forwarded to the Asset Elasticity Estimation Engine (described in the next section) which returns the available flexibility.

7.2.3 Asset Elasticity Estimation Engine

After presenting the Price Calculation Component above, we proceed with the definition of the Asset Elasticity Estimation Engine as an integral part of implicit demand response framework. Thus, the framework towards the extraction of high-level demand elasticity profiles is provided in this section. The functional role of this module is:

a) to periodically report the maximum available elasticity at portfolio level;
b) to calculate ad-hoc demand elasticity figures (day-ahead forecast of demand flexibility potential) that will further facilitate decision-making at portfolio level; i.e. demand elasticity for what-if simulation analysis engine (broadcasting pricing schemes to relevant WiseGRID tools);

In order to facilitate both of the abovementioned functional roles we define an interface which will periodically update high-level demand elasticity profiling model parameters. This is the result of the training process described in section 6.3 and is presented in the following figure.

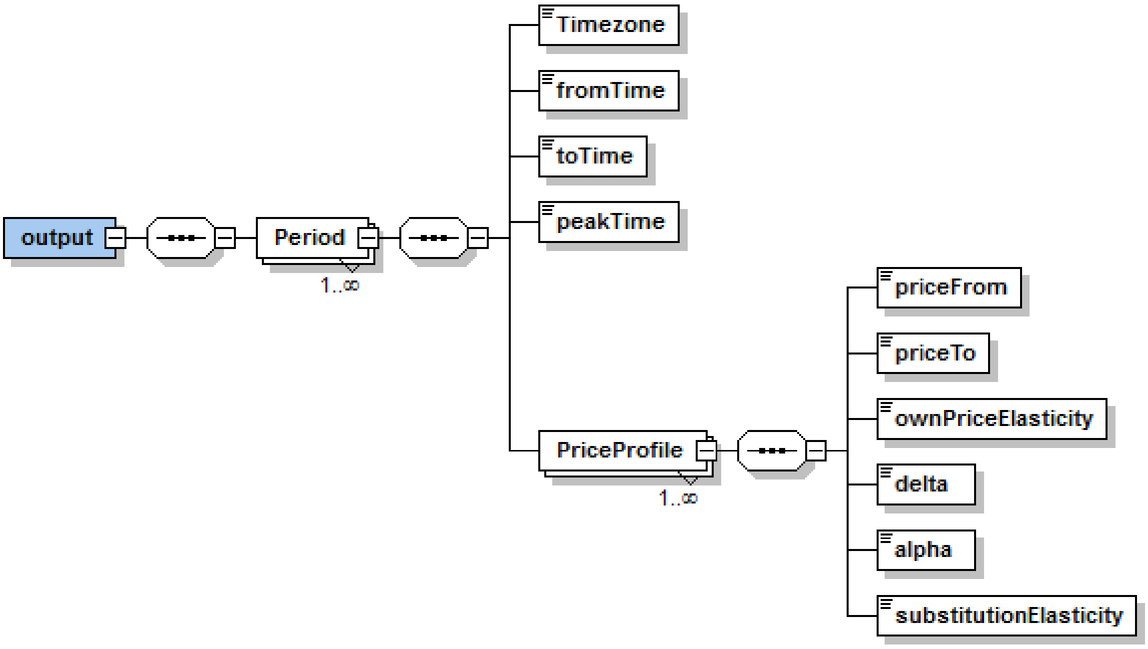

Figure 37 – Demand elasticity profiling model parameters

In addition, the JSON message to report the model parameters is specified in the following code

AMQP queue: "**postElastisticityProfile/assetID**"

With JSON message


```
{
"output": {
"Period": [
   {
    "periodId": "2",
    "Timezone": "2001-12-17T09:30:47-05:00",
    "fromTime": "00:00",
    "toTime": "01:00",
    "peakTime": "1",
    "PriceProfile": [
     {
      "priceId": "2",
      "priceFrom": {
       "inclusive": "1",
       "text": "3.14159"
      },
      "priceTo": {
       "inclusive": "1",
```

```
        "text": "4.14159"
       },
       "ownPriceElasticity": "0.314159",
       "delta": "3.14159"
     },
     {
       "priceId": "2",
       "priceFrom": {
        "inclusive": "1",
        "text": "4.14159"
       },
       "priceTo": {
        "inclusive": "1",
        "text": "5.14159"
       },
       "ownPriceElasticity": "0.34159",
       "delta": "3.14159"
     }
    ]
   },
   {
    "periodId": "2",
    "Timezone": "2001-12-17T09:30:47-05:00",
    "fromTime": "02:00",
    "toTime": "03:00",
    "peakTime": "0",
    "PriceProfile": [
     {
       "priceId": "2",
       "priceFrom": {
        "inclusive": "1",
        "text": "3.14159"
       },
       "priceTo": {
        "inclusive": "1",
        "text": "3.14159"
```

```
        },
        "ownPriceElasticity": "3.14159",
        "delta": "3.14159"
      },
      {
        "priceId": "2",
        "priceFrom": {
          "inclusive": "1",
          "text": "3.14159"
        },
        "priceTo": {
          "inclusive": "1",
          "text": "3.14159"
        },
        "ownPriceElasticity": "3.14159",
        "delta": "3.14159"
      }
    ]
  }
 ]
}
}
```

7.2.4 Maximum Elasticity Interfaces Definition

In the previous section, the building elasticity engine and its relevance to the DR optimization framework was defined. In this section, the messages and interfaces for the periodic report of high-level demand elasticity figures on each asset per time-interval based on the price calculation module described in section 7.2.2 are documented. This information is made available internally to the Retailer and is not part of the broadcasted signal of pricing levels to the WG IOP.

The following is the outcome of the process for maximum elasticity estimation for each asset:

The JSON body response (simulated) for **implicit DR** should have the following format:

```
{
 "market" : "ToU, CPP or RTP?",
 “timestampCreated” : “datetime”,
 “ParticipationList”: [
   {
```

```
    "assetId": "asset01",
    “dsmParticipation” : [
        {“interval”: "0",
         “value”: “xxxx”},
        {“interval”: "1",
         “value”: “xxxx”},
          .
         {“interval”: "23",
         “value”: “xxxx”},
    ]
  },
  {
    "assetId": " asset02",
    “dsmParticipation” : [
        {“interval”: "0",
         “value”: “xxxx”},
        {“interval”: "1",
         “value”: “xxxx”},
          .
         {“interval”: "23",
         “value”: “xxxx”},
    ]
  },...
 ]
}
```

The above represents a full list of the assets in the portfolio considering the **maximum** available elasticity (**“value”**) per interval. Thereafter, the optimization process is responsible for defining the appropriate pricing signals in order to optimally balance the associated portfolio imbalances.

7.2.5 Optimal Price Selection and Dispatch (HYP)

WiseCOOP reports the pricing scheme to all interested applications (WiseCORP, WG STaaS/VPP, WiseEVP, WiseHOME) in order to exploit it in their internal processes (energy optimization, visualization, etc.). The following message represents the pricing scheme.

while the JSON body for **broadcasting the pricing scheme** is:
{   "market": "ToU, CPP or RTP?",

"systemUser": "retailer01",
"timestampCreated": "datetime",
"dsmPriceList": [{
"interval": "0",
"value": "xxxx"
},
{
"interval ": "1",
"value": "xxxx"
},
....
{
"interval": "23",
"value": "xxxx"
}
]
}

"**interval**" represents the time interval (hour) for day-ahead application, "**value**" represents the price level for the specific interval as selected by the optimization process. After consultation with business partner, the pricing scheme should be the same for all assets as per the **fairness** assumption.

The relevant modules that comprise implicit demand response architecture along with their interfaces have been defined. Next, we define the tools used in case of near real-time demand response requests (explicit demand response). These tools accommodate the need of an Aggregator that may want to offer potential flexibility when triggered by the DSO. Such a service is offered at portfolio level by accumulating potential demand response of controllable devices at building level.

7.3 EXPLICIT DEMAND RESPONSE COMPONENTS

7.3.1 Demand Response Message Handler

As per the description in Deliverable D13.1 [22], upon forecast of a congestion problem in the grid, the DSO evaluates the necessary actions (increase/decrease of demand in a certain area of the grid) and subsequently publishes a flexibility request. The WG IOP is configured in such a way that the request reaches each one of the aggregators participating in the platform.

This message is received at the Demand Response Message Handler module and is propagated accordingly.

AMQP exchange name: "flexReqs"

Message properties:

Reply_to: "flexOffers"

Correlation_id

Message payload: FlexRequest

Section	Parameter	Description
FlexRe-quest		FlexRequest messages are used by BRPs and DSOs to request flexibility from Aggregators. In addition to one or more PTU elements with Disposition=Requested, indicating the actual need to reduce consumption or production, the message should also include the remaining PTUs for the current Period where Disposition=Available, so the receiving Aggregator can decide whether time-shifting load is an option to meet the needs of the requesting party.
	FlexRe-quest.PTU-Duration	ISO 8601 time interval (minutes only, for example PT15M) indicating the duration of the PTUs referenced in this Flex* message. Although the PTU length is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant PTU duration. The project will use PTU duration of 15 minutes.
	FlexRequest.Period	Day (in yyyy-mm-dd format) the PTUs referenced in this Flex* message belong to.
	FlexRe-quest.TimeZone	Time zone ID (as per the IANA time zone database, http://www.iana.org/time-zones, for example: Europe/Amsterdam) indicating the UTC offset that applies to the Period referenced in this message. Although the time zone is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant UTC offset. TimeZone will be a fixed value per pilot site.
	FlexRe-quest.CongestionPoint	Entity Address of the Congestion Point this Flex* message applies to.
	FlexRequest.Sequence	Sequence number of this message, which should be incremented each time a new revision of a Flex* message is sent. To ensure unique incrementing sequence numbers, use of the format yyyymmddHHMMSSssss (year, month, day, hour, minutes, seconds and milliseconds, respectively) is highly recommended.
	FlexRequest.Expira-tionDateTime	Date and time, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime) until which the Flex* message is valid. DSO will only process offers received before this expiration timestamp.
PTU		The PTU element represents one or more Program Time Units.
	PTU.Disposition	Indication whether the Power specified for this PTU represents available capacity or a request for reduction/increase (Valid value: “Requested”). WiseGRID Cockpit will only produce requests for reduction/increase of power.
	PTU.Power	Power specified for this PTU in Watts. A positive value indicates that power flows towards the Prosumer (consumption), a negative value indicates flow towards the grid (production).
	PTU.Start	Number of the first PTU this element refers to. The first PTU of a day has number 1.
	PTU.Duration	The number of the PTUs this element represents. Optional, default value is 1.

Table 11 – Flexibility Request Message used by BRPs and DSOs

Once the request sent by the DSO is received through the WG IOP, it is evaluated and processed as described in following workflow:

1. Firstly, the Aggregator filters and ranks the assets at its disposal, evaluating in this way the maximum available flexibility (section 7.3.2);

2. Then, a compensation for the provision of the requested flexibility amount is calculated, based on a price per kWh defined by the Aggregator (section 7.3.4);
3. Thereafter, a response message is sent to the DSO via the WG IOP (section 7.3.4) with a specific offer for the required flexibility;
4. Lastly, in case the DSO accepts the offer, a DR request is dispatched to the selected energy assets (section 7.3.5) (or rejects the offer in which case no further action is required).

7.3.2 Filtering and Ranking modules

Narrowing down to and ranking a selection of assets that comprise a portfolio is an important aspect of the explicit DR optimization framework. In this way portfolio-level business objectives relevant to stakeholders involved can be incorporated. In WiseGRID we highlight the importance of this component as an enabler to effective **peak-load management** and transformation of **demand-driven Virtual Power Plants** (VPPs) to active energy market commodities, competitive against traditional resources (power generation) used for the provision of balancing and ancillary services to the distribution grid.

Especially, **peak-load management and congestion management** hold a significant position in WiseGRID. By properly managing the flexibility offered by demand and optimally coordinating highly flexible portfolios, the DR optimization framework can provide:

(a) maximum penetration of RES into the energy mix,
(b) significant peak demand reduction,
(c) enhanced security of energy supply, and
(d) monetary benefits for prosumers (energy cost savings and avoidance of high energy charges during peak periods, incentives, rebates, etc.) and Aggregators (trading an inexpensive and highly competitive commodity – demand flexibility – in the balancing and ancillary services markets).

Filtering of assets that participate in an explicit DR campaign is based on:

i. geographical positioning and grid connection point; and
ii. whether the asset has an active engagement with an aggregator for participation in DR campaigns with a Service Level Agreement (SLA) that permits an additional trigger.

In this way, the relevant actor can target specific areas that are predicted to have network congestion issues and, also, perform peak-load management. After selecting the assets to be targeted (portfolio), these are then ranked based on their:

a) **maximum available flexibility**; 2-hours ahead in the case of explicit demand response;
b) **historical demand response behaviour**; this is a measure of how effectively each asset responded to past DR requests;
c) **past number of DR triggers**; in other words, this represents the number of times that each particular asset (building in this context) has been triggered in the past to perform a DR campaign.

In regards to **maximum** available flexibility, each asset is rated using the following equation:

$$J_{flex}^{asset} = \sum_{j}^{N} q(j,u) \tag{55}$$

Where j ∈ [0, N) is the time-step (interval), N is the time-horizon, u is the control variable (setpoints), and q represents the potential flexibility at time interval j and for control variable u. Note that for explicit demand response this aggregation is performed over the available devices inside the building and is subject to comfort boundary constraints as described in Chapter 6.2. Hence, a maximization of flexibility is performed for a given time horizon and an allowable list of control variables (device control actions that retain the occupant's comfort).

In respect to historical demand response of each asset, the following average ratio is used:

$$J_{DRresponsiveness}^{asset} = \frac{1}{D}\sum_{i}^{D}\frac{DR_{performed}^{asset}}{DR_{requested}^{asset}} \tag{56}$$

$DR_{requested}$ represents the historical DR signals sent to the specific asset while $DR_{performed}$ is the actual flexibility given; D is the total number of DR signals sent to the respective asset irrespective of implicit of explicit DR request. This equation is the average DR responsiveness of the asset (%).

Last but not least, the total number of DR triggers is accounted for by:

$$J_{triggers}^{asset} = D \tag{57}$$

Where D, is the total number of DR signals sent to the specific asset.

Ranking of the assets is then performed by using a weighted objective function:

$$J = w_1 * J_{flex}^{asset} + w_2 * J_{DRresponsiveness}^{asset} - w_3 * J_{triggers}^{asset} \tag{58}$$

Where the number of DR triggers acts as a penalising factor to the ranking of the asset. This aims to distribute the triggers to action among all involved buildings rather than calling upon the same buildings every time. This is fair in the sense that all buildings can reap the potential benefits, and the better ones (e.g. more reliable or performant ones) do not suffer from overuse and fatigue. Note that each objective is **normalised** over the maximum value observed in the portfolio (J/J_{max}).

Towards this direction, **comfort-based** demand flexibility is characterised for each asset and paves the way for defining and executing highly effective demand response strategies at portfolio level for the implementation of peak-load management strategies and the definition of VPP setups (utilizing the aggregated flexibility of DERs).

The functional aims of these tools are to:

1. periodically report a sorted and ranked list of assets along with their available potential flexibility (every 15 or 30 minutes),
2. act as the layer which disaggregates the DR signal requested to each asset (dispatch DR signal to each asset)

For explicit demand response, DER Models described in section 6.2 play a pivotal role in predicting the future behaviour of each device type and therefore they are useful for near-future control optimization. For the explicit demand response case, control variables are setpoints for the controllable loads (devices). The formalization of the optimization approach used is given below (within WiseCORP):

$$\begin{aligned} &\min J_k \\ &\text{s.t.} \\ &u_{min} \le u(k+j \mid k) \le u_{max} \qquad \forall\, j = 1, \ldots, N_u \\ &y_{min} \le \hat{y}(k+j \mid k) \le y_{max} \qquad \forall\, j = 1, \ldots, N_u \end{aligned} \tag{59}$$

Where, N_u is the future control horizon; $u(k + j \mid k)$ is the control signal at time k + j, computed at time k; u_{min}. u_{max} are the lower and upper control boundaries of the device (explicit DR); y_{min} and y_{max} are the lower and upper comfort boundaries learnt for the consumer (explicit DR).

The objective function J is selected as such so that to meet the requested amount of flexibility within the requested time-horizon (2-hours ahead):

$$J_k = \left(DQ_{requested} - \sum_{i}^{Devices}\sum_{j=1}^{N_u} flex(u_{ij})\right)^2 \tag{60}$$

Where i ∈ Devices and u_j is the control signal for device i at time j. Note that $flex(u_j)$ is calculated as described in section 6.2.8 for a given setpoint and device.

The goal is to control a set of devices that affect visual and thermal comfort in order to deliver a potential amount of demand flexibility within a set of thermal and visual comfort boundaries.

For the explicit DR **triggering**, the JSON representation of the message format between the Aggregator and the explicit demand response layer of WiseCOOP is presented in the following table:

The JSON body request should have the format:

{
"space": "North",
"type": "explicit",
"dsm_request": [
{
"value": "XX" or null,
"timePeriod":"120"
}
]
}

The configuration parameters required for the simulation analysis are: the **type of DR request** (directly associated with the type of contract); in WiseGRID project: "explicit" demand response type, the **region** to spatially limit the portfolio and a **list** of the **demand side request** over a time-period. **value** can be null in case a periodic report of demand flexibility is required.

For explicit the **dsm_request** is a list of **1 object** with timePeriod: "120" (i.e. minutes. This should always be *multiples of 15-minute intervals, with 120 being the maximum – 2 hours*)

Figure 38 – WiseCOOP Explicit Demand Response Interface with Aggregator

For the explicit demand response case, this message is used by the Aggregator to **trigger** explicit DR and respond to DSO's requests or request a periodic report ("value" is set to null).

7.3.3 Periodic Report of Portfolio Flexibility

In the case of periodically reporting the available flexibility, the following represents the message incorporating the maximum demand flexibility per building:

The JSON body response (simulated) for **explicit DR** should have the following format:

{
"market" : "explicit",
"timestampCreated" : "datetime",
"ParticipationList": [
{
"assetId": "asset01",
"dsmParticipation" : [
{"interval": "0",
"value": "xxxx",

```
          "cost" : "XXX"},
         {"interval": "1",
          "value": "xxxx",
          "cost" : "XXX"},
          ...
          {"interval": "23",
          "value": "xxxx",
          "cost" : "XXX"}
    ]
  },
  {
    "assetId": " asset02",
    "dsmParticipation" : [
         {"interval": "0",
          "value": "xxxx",
          "cost" : "XXX"},
         {"interval": "1",
          "value": "xxxx",
          "cost" : "XXX"},
          ...
          {"interval": "23",
          "value": "xxxx",
          "cost" : "XXX"}
    ]
  },...
 ]
}
```


The above JSON body defines the maximum demand flexibility available for the buildings included in the portfolio. We now proceed with the compensation calculation of the Aggregator and the response message to the DSO.

7.3.4 Aggregator Compensation Calculation for Portfolio Flexibility

As per the description in Deliverable D13.1 [22], each one of the aggregators will process the request, evaluate the feasibility of responding to it (accordingly to the available resources), and finally post an offer, describing up to which extent they can support the DSO, and the associated cost of that action.

The Aggregator is compensated only for the triggered assets. Hence, the ranking process outlined above defines the bid and, therefore, the set of buildings that are anticipated to be triggered. Consequently, their aggregated cost is the compensation sent back to the DSO as exemplified in the format presented in the following table per time unit.

Queue: “FlexOffers” (accordingly to the “reply_to” property of the FlexRequest)

Message properties:

reply_to: name of the queue where the aggregator expects the offer or rejection

Correlation_id: correlation id of the flex request

Section	Parameter	Description
FlexOf-fer		FlexOffer messages are used by Aggregators to make DSOs and BRPs an offer for providing flexibility. A FlexOffer message contains a list of PTUs, with for each PTU the change in consumption or production offered, plus the price for this amount of flexibility. FlexOffer messages should only be sent once a FlexRequest message has been received and must never be sent unsolicited. Note that multiple FlexOffer messages may be sent based on a single FlexRequest: for example, one offer that exactly matches the power reduction requested, plus one with a different amount of reduction, with more favourable pricing. When responding to a BRP-originated FlexRequest, an Aggregator may send an empty FlexOffer message (i.e. a message not containing any PTU elements) in order to indicate that no flexibility is available.
	FlexOffer.FlexRequest-Sequence	Sequence number of the FlexRequest message this request is based on. The combination of FlexRequestOrigin and FlexRequestSequence should be unique.
	FlexOffer.Currency	ISO 4217 code indicating the currency that applies to the prices listed for each PTU (EUR for all pilot sites)
	FlexOffer.PTU-Duration	ISO 8601 time interval (minutes only, for example PT15M) indicating the duration of the PTUs referenced in this Flex* message. Although the PTU length is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant PTU duration. The project will use PTU duration of 15 minutes.
	FlexOffer.Period	Day (in yyyy-mm-dd format) the PTUs referenced in this Flex* message belong to.
	FlexOffer.Time-Zone	Time zone ID (as per the IANA time zone database, http://www.iana.org/time-zones, for example: Europe/Amsterdam) indicating the UTC offset that applies to the Period referenced in this message. Although the time zone is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant UTC offset. TimeZone will be a fixed value per pilot site.
	FlexOffer.CongestionPoint	Entity Address of the Congestion Point this Flex* message applies to.
	FlexOffer.Sequence	Sequence number of this message, which should be incremented each time a new revision of a Flex* message is sent. To ensure unique incrementing sequence numbers, use of the format yyyymmddHHMMSSssss (year, month, day, hour, minutes, seconds and milliseconds, respectively) is highly recommended.
	FlexOffer.ExpirationDateTime	Date and time, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime) until which the Flex* message is valid. DSO will only process offers received before this expiration timestamp.
PTU		The PTU element represents one or more Program Time Units.
	PTU.Power	Power specified for this PTU in Watts. A positive value indicates that power flows towards the Prosumer (consumption), a negative value indicates flow towards the grid (production).
	PTU.Start	Number of the first PTU this element refers to. The first PTU of a day has number 1.
	PTU.Duration	The number of the PTUs this element represents. Optional, default value is 1.
	PTU.Price	The price offered or accepted for supplying the indicated amount of flexibility in this PTU.

Table 12 – Flexibility Offer message used by Aggregators to DSOs and BRPs

The DSO waits until the validity time of the request is reached and evaluates the set of received offers. Upon decision, the order/rejection to the corresponding aggregators is sent.

Queue: accordingly to reply_to parameter of the offer

Payload: FlexOrder

Section	Parameter	Description
FlexOrder		FlexOrder messages are used by DSOs and BRPs to purchase flexibility from an Aggregator based on a previous FlexOffer. A FlexOrder message contains a list of PTUs, with, for each PTU, the change in consumption or production to be realized by the Aggregator, plus the accepted price to be paid by the DSO or BRP for this amount of flexibility. This PTU list should be copied from the FlexOffer message without modification: Aggregator implementations will (and must) reject FlexOrder messages where the PTU list is not exactly the same as offered.
	FlexOrder.Currency	ISO 4217 code indicating the currency that applies to the prices listed for each PTU (EUR for all pilot sites)
	FlexOrder.PTU-Duration	ISO 8601 time interval (minutes only, for example PT15M) indicating the duration of the PTUs referenced in this Flex* message. Although the PTU length is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant PTU duration. The project will use PTU duration of 15 minutes.
	FlexOrder.Period	Day (in yyyy-mm-dd format) the PTUs referenced in this Flex* message belong to.
	FlexOrder.TimeZone	Time zone ID (as per the IANA time zone database, http://www.iana.org/time-zones, for example: Europe/Amsterdam) indicating the UTC offset that applies to the Period referenced in this message. Although the time zone is a market-wide fixed value, making this assumption explicit in each message is important for validation purposes, allowing implementations to reject messages with an errant UTC offset. TimeZone will be a fixed value per pilot site.
	FlexOrder.CongestionPoint	Entity Address of the Congestion Point this Flex* message applies to.
	FlexOrder.Sequence	Sequence number of this message, which should be incremented each time a new revision of a Flex* message is sent. To ensure unique incrementing sequence numbers, use of the format yyyymmddHHMMSSssss (year, month, day, hour, minutes, seconds and milliseconds, respectively) is highly recommended.
	FlexOrder.ExpirationDateTime	Date and time, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime) until which the Flex* message is valid. DSO will only process offers received before this expiration timestamp.
	FlexOrder.OrderReference	Order number assigned by the BRP or DSO originating the FlexOrder. To be stored by the Aggregator and used in the settlement phase.
PTU		The PTU element represents one or more Program Time Units.
	PTU.Power	Power specified for this PTU in Watts. A positive value indicates that power flows towards the Prosumer (consumption), a negative value indicates flow towards the grid (production).
	PTU.Start	Number of the first PTU this element refers to. The first PTU of a day has number 1.
	PTU.Duration	The number of the PTUs this element represents. Optional, default value is 1.
	PTU.Price	The price offered or accepted for supplying the indicated amount of flexibility in this PTU.

Table 13 - Flexibility Order message used by DSO to Aggregators

In case the DSO accepts the offer, then a DR dispatch is performed to the selected energy assets under the Aggregator. The interfaces related to this latter action are described in the next section.

7.3.5 Demand Response signal dispatch to energy assets

After filtering, ranking and optimizing for **explicit demand response** and in case the DR offer is accepted by the DSO, the asset-specific flexibility request is broadcasted by WiseCOOP to WG IOP in the following form:

The JSON body for **explicit DR broadcasting (eiEvent)** to assets should have the following format:


```
{
  "eiEventDescriptor": {
    "eventID": "1",
    "createdDateTime": "2012-12-13T12:12:12"
  },
  "eiEventSignals": {
    "eiEventSignal": [
      {
        "signalID": "17",
        "startTime": "2012-12-13T12:00:00",
        "activePeriod": "PT15M",
        "eiTarget": {
          "venID": "assetID",
          "aggregatedPnode": "8"
        }
      },
      {
        "signalID": "97",
        "startTime": "2012-12-13T12:15:00",
        "activePeriod": "PT15M",
        "eiTarget": {
          "venID": "12345",
          "aggregatedPnode": "1.0"
        }
      },
      {
        "signalID": "107",
        "startTime": "2012-12-13T12:20:00",
        "activePeriod": "PT15M",
        "eiTarget": {
          "venID": "assetID",
          "aggregatedPnode": "-0.4"
        }
      }
    ],
    "numDataSources": "3"
```

```
  },
  "activePeriod": "PT45M",
  "eiTarget": {
    "venID": "assetID",
    "aggregatedPnode": "8.6"
  }
}
```

The above JSON object is broadcasted as a list of objects, each one is dedicated to a specific asset, defined by the unique asset key (**venID**) and represents an explicit DR request to the specific asset (building).

The following message is the **respone** (**EiEventResponse)** of each asset to the DR triggering depending on whether the asset opts in or out of the DR trigger:


```
{
    "eventID": "123",
    "venID": "assetID",
    "signalIDs": ["17", "97", "107"],
    "optType": "true" OR "false"
}
```


While the following JSON object is the **reported result – actual flexibility delivered (EiEventReport)** of the explicit DR signal:


```
{
   "reportID" : "1233",
   "reportName" : "reportName1",
   "eventID" : "123",
   "venID" : "assetID",
   "aggregatedPnode" : "8.6",
   "reportedSignals" : [
        {
           "signalID" : "17",
           "aggregatedPnode" : "8"
        },
        {
           "signalID" : "97",
           "aggregatedPnode" : "1.0"
        },
        {
           "signalID" : "107",
           "aggregatedPnode" : "-0.4"
        }]
}
```


The message structure and attributes above are partly based on the OpenADR message structure [23] to foster interoperability with other OpenADR compliant tools.

The role of the WiseGRID explicit demand response framework is to trigger DR strategies for consumers/prosumers and receive the results of participation in DR campaigns.

Events are generated by WiseGRID explicit DR optimization framework and sent to the VEN (WiseCORP). If a response is required, the VEN acknowledges its opt-in or out-out disposition by responding with an *EiEventResponse element*.

EiEvent elements describe individual events, signal values, and time periods that apply to signals. Each *eiEvent* has an *eventDescriptor element* containing event information: event id and created timestamp.

The event signals that are applied over the entire active period are defined in an *EiEventSignals* element. This super-element contains one or more elements, each with a sequence of durations, the sum of which must equal the full duration of the active period. The *eiTarget* contains the value of the signal (DR flexibility requested) and the unique asset ID.

An *EiEventResponse* element contains the event id corresponding to the respective DR event and the VEN id which is the unique asset key. It also contains the list of signal ids requested (signalIDs). The optType may have a value of "true" or "false" to indicate the VENs disposition for a given event.

Furthermore, the results from DR participation (actual flexibility delivered) are reported back via the associated service (WG IOP). *EiEventReports* are published to WG IOP in order to indicate the actual flexibility delivered. The typical **JSON** body for reporting DR participation is defined above (*EiEventReport*).

Related to security, TLS must be used to encrypt all traffic regardless of the authentication method used. The client must always validate the server's TLS certificate given during the handshake. The entity initiating the request (the client) must have an X.509 certificate that is validated by the server during the TLS handshake. If no client certificate is supplied, or if the certificate is not valid (e.g., it is not signed by a trusted CA, or it is expired) the server must terminate the connection during the TLS handshake. If the certificate appears valid during the TLS handshake, the connection is established and the HTTP request proceeds. Once the server receives the HTTP request, it must perform authentication, given the credentials in the client certificate.

8 CONCLUSIONS

This document provides the design and specification of WiseGRID DR Optimization framework, as the back-end application running to support the Aggregator and Retailer business roles. In essence, this deliverable designs and provides specifications of flexibility models as well as means to exploit such flexibility models through an optimization framework; such a framework facilitates demand response strategies implementation. To this end, demand modelling approaches along with optimization of loads that take into account the demand modification potential of assets in order to participate in alternative demand response strategies with the aim of both network operation and market participation optimization are described.

The following **demand flexibility models** developed within the context of WiseGRID project were presented:

1. **Comfort-based demand flexibility model** reflecting real-time demand flexibility as a function of multiple parameters, such as time, device operational characteristics, environmental context/ conditions and individual/group occupant comfort preferences. Innovative and well proven machine-learning techniques are utilized to improve the accuracy of DER models by taking into account information related to events from user behaviour and respective comfort preferences.
2. In lack of low-level context information, high-level **Price-based demand elasticity models** are developed, reflecting temporal real-time demand elasticity as a function of multiple contextual (environmental) and market (price and incentive schemes) variables.
3. **Electric Vehicle demand flexibility model** were developed that reveal the energy needs of batteries and EVs along with their state of charge and discharging rates for appropriate flexibility provision.

The WiseGRID DR Optimization framework is also presented for two business cases; namely, explicit and implicit demand response strategies. In this way, the framework provides the supporting tools to Retailers (WiseCOOP) for implicit demand response strategies during which the Retailer can balance its own portfolio by broadcasting day-ahead dynamic tariffs to its participating assets (buildings, charging stations, etc.). Moreover, the framework enables Aggregators to exploit the full potential flexibility of their assets (WiseCOOP). Through WiseCORP, facility/building managers can participate in DR campaigns by automatically adapting setpoints for the devices of buildings in a human-centric manner, offering demand flexibility for congestion management and peak-load shedding, and are consequently compensated for it by the DSO.

The services provided by the WiseGRID DR Optimization framework meet all the applicable requirements that are outlined in Chapter 3. The following table gives an overview of the applicable requirements and breaks them down to four groups that reflect: a) Data Management, b) Demand Response Strategies accommodation, c) Demand Modelling including comfort/elasticity profiling and DER modelling, and d) general requirements.

In summary, the WiseGRID DR optimization framework integrates all the relevant components and allows communication amongst them by a data management layer in each tool; internal RabbitMQ implementations represent the data management layer in WiseCOOP and WiseCORP, while the WG IOP aids the communication between different platforms. Through these implementations, real-time environmental and operational data can be retrieved where available, as well as energy consumption monitoring of assets, meeting in this way the Data Management requirements.

Being able to monitor and store real-time data facilitates the extraction of accurate comfort and elasticity profiling and DER models for the respective assets (devices, batteries, etc.). Hence, the requirements that are classified in Demand Modelling group are also taken into full consideration.

Last but not least, the WiseGRID DR optimization framework, incorporates information from data monitoring and demand modelling, and sets forth the way for implementing novel Demand Response Strategies; accommodating in this way the requirements described in the following table.

In the following list it is possible to see the requirements needed for the well-functioning of the DR framework.

Requirement ID	Description	Classification	Priority
DRF_003	The user needs to be able to configure the electricity tariff, or connect it with some Public API in case of real-time pricing	Data Management	✓
DRF_004	Energy Storage should be used in order to provide flexibility to the DR	Demand Modelling	✓
DRF_005	The system should be compatible with others at the project in order to be able to share information	Data Management	✓
DRF_006	Different types of demand flexibility profiles will be defined as part of the consumer-centric DR profiling addressing the objectives of the project	Demand Modelling	✓
DRF_007	The comfort-based demand flexibility profiles should be designed taking into account remote monitoring (and controllable) of building loads examined in the project	Data Management	✓
DRF_008	As part of comfort-based demand flexibility, we should address comfort profiles associated with the operation of energy-hungry HVAC devices	Data Management	✓
DRF_009	Towards the extraction of visual comfort profiles, information about luminance levels (luminance sensors) under different operational conditions(lighting device status) is required	Data Management	✓
DRF_010	Towards the extraction of thermal comfort profiles, information about thermal context (temperature & humidity sensors) under different operational conditions (HVAC device status) is required	Data Management	✓
DRF_011	Towards the extraction of HVAC demand flexibility profiles, information about operational conditions (HVAC device status) and HVAC energy consumption is required	Data Management	✓
DRF_012	Towards the extraction of Lighting demand flexibility profiles, information about operational conditions (Lighting device status) and energy consumption is required	Data Management	✓
DRF_014	The extraction of comfort-based flexibility profiles should be based on accurate DER models	Demand Modelling	✓
DRF_015	Towards the extraction of comfort-based demand flexibility profiles, information about energy cost (retailer tariffs) is required	Data Management	✓
DRF_016	Comfort-based demand flexibility profiles shall support the implementation of demand shifting strategies (P2H flexibility profiling extraction)	Demand Response Strategies	✓
DRF_017	Comfort-based flexibility profiles should ensure the minimum of occupants disturbance on building environment	Demand Modelling	✓
DRF_018	Comfort based Flexibility Profiles should be exploited towards the implementation of automated DR strategies	Demand Response Strategies	✓
DRF_019	Price based Flexibility Profiles should be defined, reflecting the enrolment of prosumers on price based DR scenarios	Demand Modelling	✓
DRF_020	High-level Demand Elasticity Profiles should be provided in lack of low level information (device level) information	Demand Modelling	✓
DRF_021	Towards the extraction of price based flexibility profiles, information about market prices (real-time hourly prices, day-ahead hourly prices, pricing schemes) is required	Data Management	✓

DRF_022	Towards the extraction of price based flexibility profiles, information about external weather conditions should be available	Data Management	✓
DRF_023	Towards the extraction of price based flexibility profiles, information about individual consumer consumption is required	Data Management	✓
DRF_025	A central data management unit should be responsible for capturing real-time and historical information required for the extraction of the different profiling types	Data Management	✓
DRF_026	Real-time information required for the extraction of (comfort-based, price based) Demand Flexibility profiles, should be available in real-time through an automated way	Data Management	✓
DRF_027	The consumer-centric DR profiling is running as a standalone service calculating the amount of potential flexibility at each demand side end point	Data Management	✓
DRF_028	An Advanced Flexibility Analysis component should be designed to provide analytics over demand flexibility providing assets	Demand Modelling	✓
DRF_029	The Advanced Flexibility Analysis should exploit the results from consumer-centric DR profiling engine	Data Management	✓
DRF_030	Sample analytics over the streams of flexibility data (aggregation, filtering & clustering) will be supported by the Advanced Flexibility Analysis engine	Data Management	✓
DRF_031	Input values (capacity, response capability, location, time) will set the configuration parameters for the analytics process	Data Management	✓
DRF_032	Along with real-time analytics, short term forecasting of demand flexibility should be provided by the Advanced Flexibility Analysis engine	Data Management	✓
DRF_033	The outcomes of Advanced Flexibility Analysis engine may be available for visualization or to a DSS for DR strategies implementation at consumers level	Data Management	✓
DRF_034	An Optimization DSS component should be designed to enable the aggregation of multiple consumers to participate in DSM strategies	Demand Response Strategies	✓
DRF_035	The Optimization DSS component should be designed to allow for the selection of the appropriate aggregated demand side assets to participate in DR programs	Demand Response Strategies	✓
DRF_036	The Optimization DSS component should enable interacting with different grid and market stakeholders requesting demand flexibility for the business services	Data Management	✓
DRF_037	The Optimization DSS component should take into account the different DR contracts towards the selection of customers to participate in the associated campaigns	Data Management	✓
DRF_038	The Optimization DSS component should be designed to dispatch the DR signal to the different demand side end points	Data Management	✓
DRF_039	The Optimization DSS component should be designed to dispatch the associated DR signal by taking into account the DR Contract	Data Management	✓
DRF_040	The Optimization DSS component should estimate the impact of DR strategies to the active consumers, by taking into account the outcomes from consumer-centric DR profiling engine	Data Management	✓
GEN_005	WiseGRID must promote a 'level playing field' which does not discriminate between competitors (e.g., suppliers, aggregators) as well as flexibility solutions (e.g., storage, DR, EVs)	General Requirements	✓

GEN_006	WiseGRID must make use of existing standards or standards under development to provide easier access to market and the dissemination of the resulting solutions worldwide	Data Management	✓

Table 14 – Fulfilment of Requirements

Following the development of the different demand flexibility services, the next step is the integration and lab-testing of these in Deliverable 14.2” WiseGRID integrated ecosystem Lab testing”.

9 REFERENCES AND ACRONYMS

9.1 REFERENCES

[1] WiseGRID Consortium, “D2.1 WiseGRID requirements, Use cases and pilot sites analysis,” 2017.

[2] WiseGRID Consortium, “D7.1 WiseCOOP and WiseCORP Apps Design,” 2018.

[3] “Achahome,” [Online]. Available: https://www.achahome.com/inverter-ac-advantages-cost-savings.html.

[4] “Air Ace Airconditioning,” [Online]. Available: http://airaceairconditioning.co.za/difference-between-an-inverter-and-non-inverter/.

[5] Kamgarpour, Maryam, Christian Ellen, Sadegh Esmaeil Zadeh Soudjani, Sebastian Gerwinn, Johanna L. Mathieu, Nils Müllner, Alessandro Abate, Duncan S. Callaway, Martin Fränzle, and John Lygeros, “Modeling options for demand side participation of thermostatically controlled loads,” 2013.

[6] Kundu, S. B. S., Sinitsyn, N. and Hiskens, I., “Modeling and control of thermostatically controlled loads,” Stockholm, Sweden, 2011.

[7] Caicedo D. Pandharipande A., and Willems F. M. J., “Daylight-adaptive lighting control using light sensor calibration prior-information,” vol. 73, no. 105 - 114, 2014.

[8] ANSI/ASHRAE, “Standard 55-2010,” 2010.

[9] Fanger, P.O., Analysis and Applications in Environmental Engineering, New York: McGraw-Hill Book Company, 1970.

[10] Roisin, B., Bodart, M., Deneyer, A., & D’herdt, P., “Lighting energy savings in offices using different control systems and their real consumption,” vol. 40, no. 4, 2008.

[11] Galasiu, A. D., & Veitch, J. A., “Occupant preferences and satisfaction with the luminous environment and control systems in daylit offices: a literature review,” vol. 38, no. 7, 2006.

[12] D. &. M. N. Lindelöf, “Bayesian estimation of visual discomfort,” vol. 36(1), 2008.

[13] N. A. Korb K, Bayesian Artificial Intelligence, Chapman and Hall, 2014.

[14] P. L. &. K. A. Davies, “Densities, spectral densities and modality,” 2004.

[15] CEN/TC, “Sustainability of construction works”.

[16] J.R. Schofield, R. Carmichael, S.Tindemans, M. Bilton, M. Woolf and G., “SmartMeter Energy Consumption Data in London Households,” [Online].

[17] D. Ton, M.A. Biviji, E. Nagypal and J. Wang, “Tool for determining price elasticity of electricity demand and designing dynamic price program,” *Innovative Smart Grid Technologies (ISGT),* 2013.

[18] A. Faruqui, and S. Sergici, “Dynamic pricing of electricity in the mid-Atlantic region: econometric results from the Baltimore gas and electric company experiment.,” *Journal of regulatory economics, vol.40, no.1, pp.82–109, Springer,* 2011.

[19] C. R. Associates, “Impact Evaluation of the California Statewide Pricing Pilot,” 2005. [Online].

[20] Li, N., Chen, L., & Low, S. H, “Optimal demand response based on utility maximization in power

networks," *In Power and Energy Society General Meeting, 2011 IEEE (pp. 1-8). IEEE.,* 2011, July.

[21] Freire, Roberto Z., Gustavo HC Oliveira, and Nathan Mendes, "Predictive controllers for thermal comfort optimization and energy savings.," vol. 40, no. 7, 2008.

[22] WiseGRID Consortium, "D13.1 WiseGRID Cockpit Design.," 2018.

[23] T. O. Alliance, "The OpenADR Primer: An Introduction to Automated Demand Response and the OpenADR Standard," 2011.

9.2 ACRONYMS

Acronyms List	
BMS	Building Management System
BRP	Balancing Responsible Party
CP	Consortium Plenary
DER	Distributed Energy Resource
DM	Dissemination Manager
DoW	Description of Work
DR	Demand Response
DSM	Demand Side Management
DSO	Distribution System Operator
EM	Exploitation Manager
EV	Electric Vehicle
HVAC	Heating, Ventilation and Air Conditioning
IPR	Intellectual Property Rights
OBIS	Object Identification System
PC	Project Coordinator
PDF	Probability Density Function
PSC	Project Steering Committee
PTU	Power Transfer Unit
PPR	Project Periodic Report
QM	Quality Management
QR	Quarterly Report
RM	Risk Management
SLA	Service Level Agreement
SVN	Subversion
TLS	Transport Layer Security
TM	Technological Manager
VEN	Virtual End Node

Table 15 – List of Acronyms

10 ANNEX A

10.1 DEVICE AND SENSOR CONFIGURATION PARAMETERS

Wise Grid configuration parameters of WiseCORP devices

Below are examples of the meta-data that would be stored in the smart device or WiseCORP.

The following is a general configuration file that includes information on all devices installed in premises .

After the following unified JSON body (in which some parts are abstractly defined), we specifically define the template (configuration JSON body) for each device and **combo** sensors (luminance+temperature).

****

JSON Config File

```
{
 "WiseCORPConfig": {
  "Controllers": {
   "SHIC": [{
    "asset id": "String",
    "shic id": "String",
    "shic obis code": "String",
    "metadata": { "type": "\"String\"" },
    "control_type": "SG-ready / serial / modbus / IR",
    "submeter": {
     "present": "true/false",
     "Splug or SLAM obis code": "String",
     "settings": {
      "LP_reporting_interval": "integer",
      "LP_resolution": "integer"
     },
     "temp_sensor": {
      "temp obis code": "string",
      "offset": "float"
     }
    }
   },...],
   "Smartplug": [{
```

```
      "asset id": "String",
      "splug id": "String",
      "splug obis code": "String",
      "nominal_power": "watts, integer",
      "submeter": {
        "settings": {
          "LP_reporting_interval": "integer",
          "LP_resolution": "integer"
        }
      }
    },...],
    "LED_lamp": [{
      "asset id": "String",
      "led id": "String",
      "led obis code": "String",
      "nominal_power": "”watts, integer”"
    }
  },...],
  "Sensors": {
    "lux_sensor": [{
      "asset id": "String",
      "lux id": "string",
      "lux obis code": "String",
      "lux_reporting_interval": "integer",
      "lux_threshold": "”lux, float”",
      "temp_sensor": {
        "temp obis code": "string",
        "offset": "float"
      },...]
    }
  }
 }
}
```

EACH TEMPLATE SEPARATELY

SHIC template

Database: ASSET01

Collection: **???**

Document: SHIC_ID?

```
{
        “smx id”: ASSET01,
        “shic id”:SHIC01,
        “shic obis code”:  "0-1-160-7-0-1"
        “type”: “SHIC”,
        “appliance metadata”:
        {
                “type”: <HVAC / AC>
                “category”:<inverter/non inverter, ID>
                “nominal power”:<watts, integer>
                “cooling capacity”:<watts, integer>
                “heating capacity”:<watts, integer>
                “cooling efficiency”:<percentage, float>
                “heating efficiency”:<percentage, float>
                “min setpoint”:<temperature value, float>
                “max setpoint”:<temperature value, float>
        },
        “control type”: <SG-ready / serial / modbus / IR>,
        “submeter”: <true/false>,
        “splug obis code”: “0-1-165-7-0-1”,
        “submeter settings”: {
                “LP reporting interval”: 60 sec,
                “LP resolution”: 60 sec,
        },
        “temp obis code”: "0-1-96-9-0-1" or “null”
        “temp sensor offset”: <float>
}
```

#Comments: The load threshold can be set to get a specific notification on specific load changes.

Lux sensor template

```
{
```

```
        "asset id": "ASSET01",
        "lux id": "LUX01",
        "lux obis code": "0-1-151-7-0-1",
        "type": "lux sensor",
        "lux reporting interval": <60 / 120 / xyz sec>,
        "lux threshold":<lux, float>
        "temp obis id": "0-1-96-9-0-1" or "null",
        "temp sensor offset": <float>
}
```

#Comments: The lux threshold can be set to get a specific notification on when the light changes. Please note that the battery powered lux meter will only report once per minute. If a specific test scenario calls for more frequent reports, the interval can be further shortened (at the obvious cost of shorter battery life time, but for test scenarios this might still be desired).

Smartplug template

```
{
        "asset id": "ASSET01",
        "splug id": "SPLUG01",
        "splug obis code": "0-1-165-7-0-1",
        "type": "smartplug",
        "nominal power":<watts, integer>
        "submeter settings": {
                "LP reporting interval": 60 sec,
                "LP resolution": 60 sec,
        }
}
```

#Comments: The load threshold can be set to get a specific notification on specific load changes.

LED lamp template

```
{
        "asset id": "ASSET01",
        "led id": "LED01"
        "led obis code": "0-1-163-7-0-1"
```

```
        “type”: “LED_lamp”,
        “nominal power””:<watts, integer>
}
```

SLAM template

```
{
        “asset id”: “ASSET01”,
        “slam id”: “SLAM01”
        “slam obis code”: "0-1-165-7-0-1"
        “type”: “smartmeter”,
        “nominal power”:<watts, integer>
        “submeter settings”: {
                “LP reporting interval”: 60 sec,
                “LP resolution”: 60 sec,
        }
}
```