D7.5 Lessons learned from use case NL#1
Version 1.0

Deliverable D7.5

30/06/2019

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement n°731289

ID & Title :	D7.5 Lessons learned from use case NL#1.		
Version :	V1.0	**Number of pages :**	50

Short Description

Deliverable 7.5 focuses on use case 1 of the Dutch InterFlex demo.

Use case 1 describes the Smart Storage Unit that is valued with the support of all the players involved: the Transport System Operator, the Distribution System Operator, the storage operator and the prosumers. It demonstrates the applicability of large-scale centralized storage units at the substation or street level for demand side management. The deployed capacity of the smart storage unit is 250 kW / 315 kWh.

To enable interaction between actors, markets and local resources a Local Infrastructure Management System (LIMS) is developed as a local interface platform from and to the potential flexibility sources.

Revision history

Version	Date	Modifications' nature	Author
V0.1	31-10-2018	Concept	Marcel Willems
V0.2	1-2-2019	Writing Chapter 3	Kees van Zwienen (Enexis), Joost Laarakkers (TNO), Wilco Wijbrandi (TNO) Beatrix Bos (TU-E), Marcel Willems (Enexis), Rob Roodenburg (CW&D), Stefan Custers (CW&D), Olga Westerlaken (Enexis), Kurt van Doren (Enexis), Sharmistha Bhattacharyya (Enexis)
V0.3	1-4-2019	Review version chapter 2	Kees van Zwienen (Enexis), Joost Laarakkers (TNO), Wilco Wijbrandi (TNO) Beatrix Bos (TU-E), Marcel Willems (Enexis), Rob Roodenburg (CW&D), Stefan Custers (CW&D), Olga Westerlaken (Enexis), Kurt van Doren (Enexis), Sharmistha Bhattacharyya (Enexis)
V0.4	9-5-2019	Final review	Joost Laarakkers (TNO), Marcel Willems (Enexis), Stefan Custers (CW&D), Olga Westerlaken (Enexis), Rik Fonteijn (TU-E), Sharmistha Bhattacharyya (Enexis), Daphne Geelen (Enexis)
V0.5	23-5-2019	Overall review	Marcel Willems (Enexis), Marianne Postmus (Enexis)
V1.0	24-6-2019	Final version	Marcel Willems (Enexis), Marianne Postmus (Enexis)

<table>
<tr><td colspan="6">Accessibility</td></tr>
<tr><td>☒Public</td><td colspan="2">☐ Consortium + EC</td><td>☐ Restricted to a specific group + EC</td><td colspan="2">☐ Confidential + EC</td></tr>
<tr><td colspan="6">Owner/Main responsible</td></tr>
<tr><td>Name(s)</td><td colspan="3">Function</td><td>Company</td><td>Visa</td></tr>
<tr><td>Marcel Willems</td><td colspan="3">Project leader</td><td>Enexis</td><td></td></tr>
<tr><td colspan="6">Author(s)/contributor(s): company name(s)</td></tr>
<tr><td colspan="6">Kees van Zwienen (Enexis), Joost Laarakkers (TNO), Wilco Wijbrandi (TNO) Beatrix Bos (TU-E), Marcel Willems (Enexis), Rob Roodenburg (CW&D), Stefan Custers (CW&D), Olga Westerlaken (Enexis), Kurt van Doren (Enexis), Sharmistha Bhattacharyya (Enexis), Daphne Geelen (Enexis) and Marianne Postmus (Enexis)</td></tr>
<tr><td colspan="6">Reviewer(s): company name(s)</td></tr>
<tr><td colspan="5">Company</td><td>Name(s)</td></tr>
<tr><td colspan="5">E.ON</td><td>Jörgen Rosvall</td></tr>
<tr><td colspan="6">Approver(s): company name(s)</td></tr>
<tr><td colspan="5">Company</td><td>Name(s)</td></tr>
<tr><td colspan="5">Enedis</td><td>C. Dumbs</td></tr>
<tr><td>Work Package ID</td><td>WP 07</td><td colspan="2">Task ID</td><td colspan="2">T7.5</td></tr>
</table>

Disclaimer: This report reflects only the author's view and the Agency is not responsible for any use that may be made of the information it contains.

EXECUTIVE SUMMARY

This report focuses on use case 1, work package 7 from the Dutch Interflex project team. It describes the components built and the first results of the field test. This deliverable is one of the six WP7 deliverables for the Horizon 2020 project Interflex.

Use case 1 involves the systems and interactions necessary to achieve a stable grid through flexibility that can be provided by stationary batteries and rotating PV. The infrastructure and chain from mentioned assets to distribution system operator (DSO) are described in this deliverable.

To arrange an optimal activation of flex systems, Local Infrastructure Management System (LIMS) and Flexibility Aggregation Platforms (FAPs), and interactions were implemented.

The different assets are controlled and connected to the LIMS. In the Dutch demo the experiences with the battery, in this report called Smart Storage Unit (SSU), are not good. We experienced a lot of trouble with delivery, testing and operation of the SSU. This has its drawback on the test results.

The connection of the other systems like rotating PV, solar car and the CroonWolter&Dross office building was no problem and worked well.

The first results of the field test show that flex order and flex supply with the battery can normally be a match. This in relation to the PV system.

Recommendations:

The complexity in integrating different systems with different protocols and interfaces was and is the big challenge for this use case. Good and basic specification are the key words. For the LIMS connection with EFI this works fine. The protocol proves his usability in this. The complexity of the battery is the weak spot. The battery in this project isn't able to meet the expectations yet and by specifying the battery it should be clear what purpose the battery should serve in the network. Make it not too complex.

Overall testing between several partners means also a lot of coordination. The project method of Agile used here proofs to be a good one. But this also means a lot of discipline by the different partners.

TABLE OF CONTENT

LIST OF FIGURES

1. INTRODUCTION

1.1. What is energy flexibility and why is it important?

Smart grids rely on flexibility (or flex), to prevent congestion on the grid, in energy production and/or consumption to compensate for the increasing numbers of renewable energy sources that are far less predictable and controllable than traditional power plants. The required flexibility ultimately has to come from smart devices in households, small medium enterprises, office buildings, electric vehicles (EVs), storage, etc.

Energy flexibility can be defined as a power adjustment sustained at a given moment, for a given duration, from a specific location within the network.

This flexibility may be used by third parties to help alleviate imbalance or congestion. Third parties will use different incentive schemes to unlock the flexibility potential, such as time of day pricing, real time pricing and feed in tariffs. These incentives should somehow be matched to the possibilities of smart devices to deliver energy flexibility.

1.2. Scope of the document

The information and views set out in this report are those of the authors and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.

This deliverable is one of the six deliverables for the Horizon 2020 project Interflex Dutch Pilot and describes the components that were built, lessons learned during the practical implementation, issues in the first two and half year of the project and first results from field-testing.

This document focuses on use case 1, describing the smart storage unit (SSU) and the technical aggregator system the 'Local Infrastructure Management System' (LIMS). Additionally, other flexibility-providing assets connected to the LIMS are described in this deliverable.

Deliverable 7.6 has the same purpose as this document, with use case 2 and 3 as scope. Deliverable 7.6 is focusing on electrical vehicles connected to the technical aggregator system the Charge Point Management System (CPMS) as flexibility asset and the technical framework for trading flexibility between multiple aggregators and the DSO.

A more in-depth specification of the several system components and the functions of Interflex in the Netherlands is described in deliverables 7.1, 7.3 and 7.4.

In the last Work Package 7 deliverable, deliverable 7.7, the results are presented from the field test scenarios that are performed throughout 2019. These field test scenarios are based on the KPIs that are defined in the project (see deliverable 7.3). The purpose of the field test is to validate the market model for flexibility delivery for the DSO.

See also figure 1 for the scope and content of the different deliverables as mentioned above.

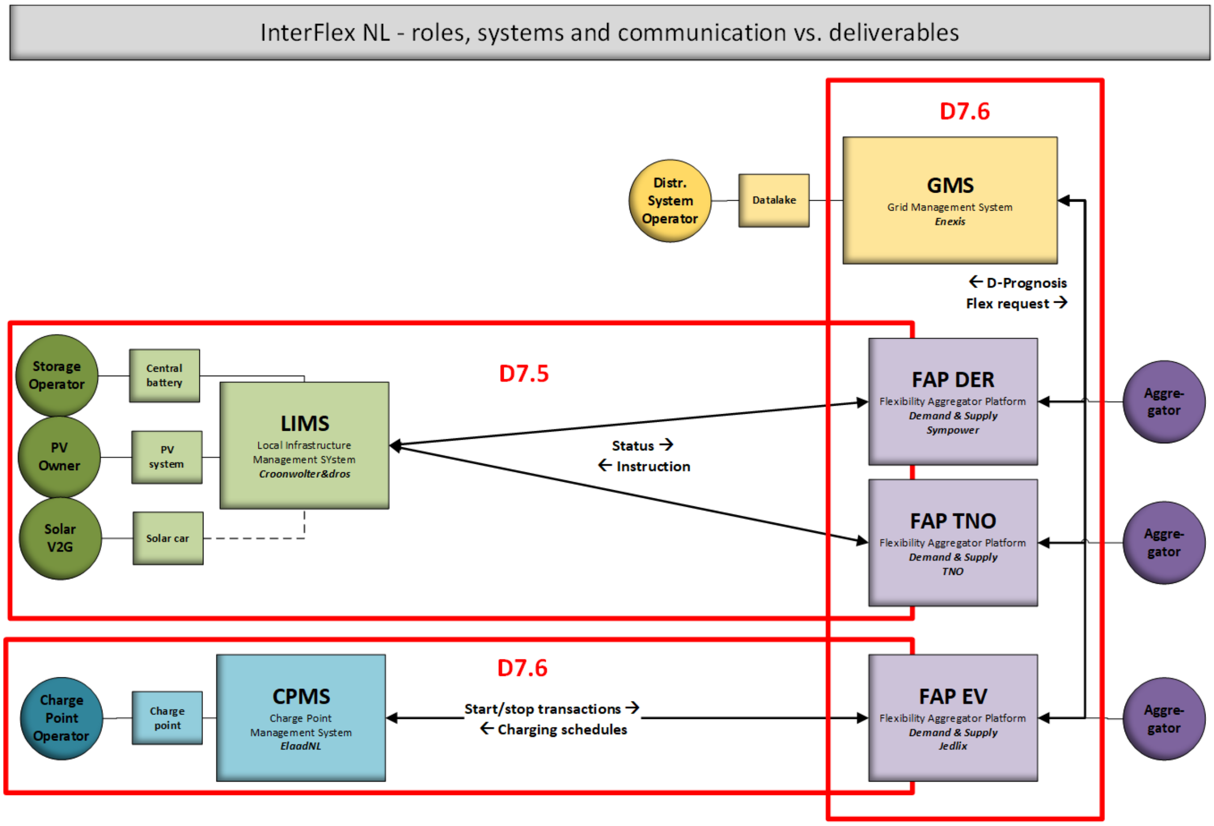

Figure 1 - Roles, systems and communication in relation to the deliverables

1.3. Notations, abbreviations and acronyms

The table below provides an overview of the notations, abbreviations and acronyms used in the document.

CA	Commercial Aggregator
CPMS	Charge Point Management System
CPO	Charge Point Operator
CS	Charging session
DSO	Distribution System Operator
ESCO	Energy Service Company
EC	European Commission
EC-GA	European Commission Grant Agreement
EFI	Energy Flexibility Interface
EFI+	Energy Flexibility Interface extend version, designed for Interflex
eMSP	eMobility Service Provider
ESCo	Energy Service Company
EU	European Union
EVSE	Electric Vehicle Supply Equipment/ ChargePoint
FAP	Flexibility Aggregation Platform
GA	General Assembly
GWP	General Work Package
KPI	Key Performance Indicator
LA	Local Aggregator
LIMS	Local Infrastructure Management system
PC	Project Coordinator
PTU	Programmable Time Unit
PV	Photovoltaic
RFID	Radio-frequency identification
SC	Steering Committee
SME	Small and medium-sized enterprises
SSU	Smart Storage Unit
TC	Technical Committee
TD	Technical Director
TSO	Transmission System Operator
UI	User Interface
USEF	Universal Smart Energy Framework
USEF+	Adapted version of USEF, designed for Interflex
WP	Work Package
WPL	Work Package Leader

Figure 2 - List of acronyms

2. IMPLEMENTATION AND CHAINTEST EXPERIENCES

This chapter describes the implementation of use case 1.

2.1. Use case 1

The goal of this use case is to validate technically, economically and contractually the usability of a smart storage unit embedded as a commercial storage. Centralised storage must be valued with the support of all the players involved: the Transport System Operator, the Distribution System Operator, the storage operator and the prosumers. It demonstrates the applicability of large-scale centralized storage units at the substation or street level for demand side management. The deployed capacity of the smart storage unit is 250 kW / 315 kWh.

To enable interaction between actors, markets and local resources a Local Infrastructure Management System (LIMS) is defined. The goal is to develop a local interface platform from and to the potential flexibility sources.

2.2. LIMS architecture

The LIMS consist of the following technical functions:

- Realise an interface from and to the smart storage unit, The CroonWolterandDross office building, solar car and PV installation
  - Collect and send measurement data from smart storage unit and PV installation for the purpose of congestion management
  - Collect and send measurement data from smart storage unit and PV installation for the purpose of voltage management
  - Monitor and maintenance of the smart storage unit
  - Operation of the smart storage unit
  - Controlling the PV panels
  - Power quality improvement
- Implementation conform standardised protocols for the interaction between commercial aggregators.

The LIMS has the following organisational functions:

- Provide local flexibility sources to the commercial aggregators
- Technically and organisationally responsible for (the interface with) these flexibility sources
- Provide (forecast)data for the negotiate of contractual agreements between aggregators for the provision of flexibility

The LIMS contains four operational software services:

- Modbus software interface – aims to communicate with DER trough the Modbus protocol and logs DER parameter data event time-based to central Enexis and LIMS database
- DER windows service – retrieves time-based (PTU) data based on the content of the EFI+ messages to control the DER (control tasks) from the LIMS database.

- LIMS windows service – Communicates with FAPs by EFI+ messages that are sent and received through web socket technology. Extracts these messages to relational database structure and logs to the central Enexis database. The raw EFI+ messages will also be logged in central Enexis database.
- Forecast windows service – Aims to forecast the office building consumption as well as the PV production. This information will also be sent to the central Enexis database and send to the FAP through the LIMS Windows service.

The LIMS-system contains a relational database (LIMS database) in which data is stored in a structured way. WebSocket technology is used for the interaction with the Flexible Aggregator platform (FAP) in combination with xml messages (EFI+). The office building, SSU and the PV-system interact through the Modbus protocol; for the forecast windows service, data concerning the weather and irradiance are required. Both will be imported to the forecast windows service via an API in combination with data from a local weather station which is connected through the DER Windows service. Data will be sent to the central Enexis database regarding the DER, EFI+ messages and forecasting data. Each DER can only be connected to one FAP.

In Figure 3, the interaction between the LIMS-components and the external components is displayed.

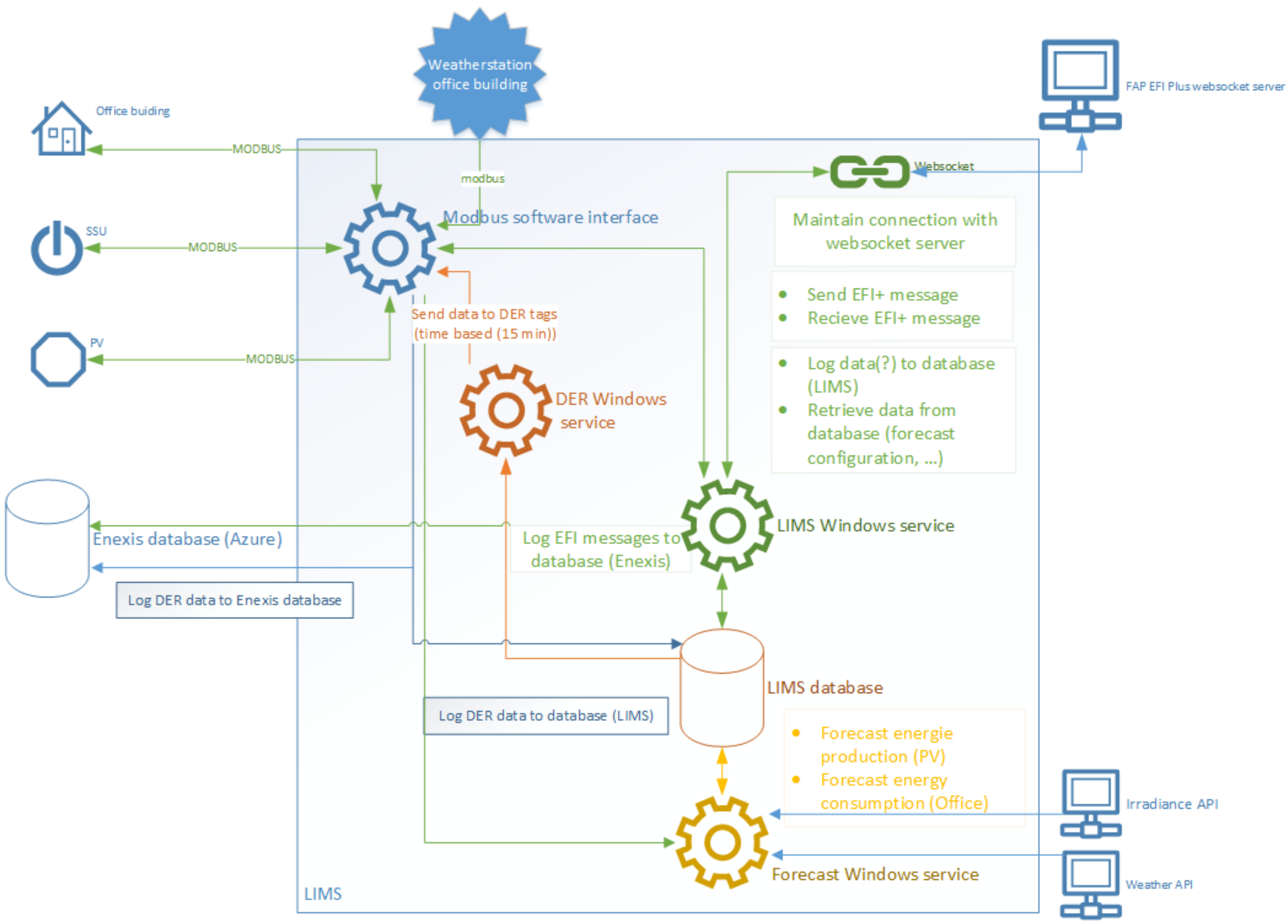

Figure 3 - Architecture of the LIMS-system

Experiences LIMS architecture

- The chosen architecture to develop the software via proven software components proved itself. The software is stable and is future proof
- The chosen database solution and interface software for logging the measurements parameters of the DER's needs also to be able to buffer the data locally if there is no database connection.
- An overview of the services that interact with the DER's, with their status would be of added value.

2.3. Modbus interface LIMS

A standard protocol is used for interfacing the LIMS to the DER: Modbus TCP, secured with the direct virtual private network (VPN) connection and encrypted with IPSec.

In order to connect the remote DER, a Virtual Private Network (VPN) is established, through a 4G router - connection. This connection is established by a hardware device called a "Tosibox", as can be seen in Figure 4.

Figure 4 - Tosibox

2.4. EFI interface implementation

What is EFI?

As of yet there was no standard interface to describe and control the energy flexibility of smart devices. The Energy Flexibility Interface, for short EFI, fills this gap and is specifically designed as a standard communication method between smart devices and Demand-Side Management (DSM) solutions. EFI is developed by TNO (the biggest Dutch independent research institute) to deal with interoperability issues encountered while experimenting and researching energy flexibility in field trials. It is a key enabling technology for the widespread deployment and adoption of Demand-Side Management to exploit energy flexibility. The interface specification of EFI is open and freely available via the Flexiblepower Alliance Network (FAN) foundation (https://flexible-energy.eu/).

Changes made to EFI in the InterFlex project: EFI+

The goal of the InterFlex project is to develop the technical conditions for an open market for providing energy flexibility services. An interface for connecting *Distributed Energy*

Resources (DERs) with an Aggregator are an important component of this goal. EFI was selected as an initial version of this interface. Where EFI would fulfil the requirements of the project it should be used as-is, but when additional requirements emerge, functionality should be added. The new and improved version of EFI is being referred to as *EFI+*

Context

EFI was developed as the interface between a device (represented by a software component referred to as the *Resource Manager*) and the *Customer Energy Manager* (CEM). The CEM is a software component managing the energy behaviour of DERs on the premise. This means that all EFI communication would only take place on the premise; behind the meter. The CEM is responsible for communication with the outside world.

In the InterFlex project however, EFI is used in a slightly different context. It is used for the communication between Commercial- and technical Aggregator. This means EFI communication now takes place outside of the premise. Although this is a small difference, it results in some additional requirements, such as information regarding the location of the DER, in order for the commercial Aggregator to be able to provide congestion management services.

Process

EFI + was initially designed by project partners without compatibility with EFI in mind. This resulted in an interface description that addressed certain issues in different ways than EFI did. However, to validate concepts in EFI, and make it easier for the implementations to move to EFI in the future, it was decided to make EFI + as similar as EFI as possible. Although this move required extra work, the advantage is that it was very easy to obtain a list of all the potential differences between EFI and EFI +. With this list, it was easy to determine if required functionality could be achieved by EFI (but in a different way), or that functionality needed to be added to EFI. After this analysis we were able to create a modified version of EFI +, which remained stable throughout all the integration tests. Quite some required functionality could be achieved using existing EFI concepts, which indicates that EFI is already fairly complete in the supported functionality.

Additions to EFI +

Some of the added functionality to EFI is specific for the use case in the InterFlex project because the scope of the EFI usage is slightly different. Some other functionality however, turned out to be relevant for the scope of EFI in general. Functionality that is not specific for InterFlex will be considered to be added to the EFI protocol; if and when this happens however depends on the standardization process EFI is currently part of.

It should be noted that some of the functionality which is outside of the scope of EFI, can still be relevant in another context. This information will then be added through an additional protocol, instead of adding it to EFI +; the pragmatic option chosen in this project.

New functionality EFI	InterFlex specific?
Providing information about the location in the grid (ConnectionID)	Yes
Providing a schedule for when the device will be unavailable for energy management in the future (e.g. due to planned maintenance)	No
Indicating why a device is unavailable (e.g. due to maintenance or due to errors)	No
Scheduling changes in the capacity of the storage unit. This is a result of the storage unit being divided into two virtual batteries, each with their own purpose. the amount of available capacity for each virtual battery can be reconfigured. The moment that this reconfiguration occurs can be important for trading on day-ahead markets	Yes
Curtailing the production from solar panels as a percentage (in contrast to curtailing based on absolute power values)	No
Providing the nominal power (maximum production and maximum consumption) for curtailable units	No

EFI over WebSockets

The EFI protocol is a protocol where two parties exchange XML messages. EFI however, does not define which mechanisms should be used to exchange these messages. In the InterFlex project, it was decided to use WebSockets for transporting the XML messages. In order to successfully implement this, an additional specification was developed for using EFI over WebSockets. This specification defines how to initiate a connection, how to deal with different types of errors, what kind of security measures should be taken and what should be done if the connection is lost.

Experiences EFI TNO

From the perspective of EFI, the InterFlex project has brought EFI the following:

- Further validations of many existing EFI concepts, with new DERs and new Aggregator platforms
- Useful recommendations for additional functionality
- A specification on how to use EFI over WebSockets
- Additional examples and documentation developed in the process

Experiences EFI Sympower:

- It's key to work with open standards from a replicability and scalability perspectives and it was good to be able to draft the required improvements from the beginning of the project (EFI+).
- EFI was built in a very modular way which allows a lot of flexibility, but the drawback is that it sometimes makes software development more complicated than needed
- Implementation of advanced features (intra-day or variable capacity) have to be taken into account already in the first phase. We noticed that implementation of those features in a later stage was a bottleneck because the interfaces need to be adjusted.
- From a scalability perspective, having one WebSocket per DER is not necessarily recommended. Sympower's suggestion would be to have one WebSocket per LIMS-FAP partnership.

- We have put into a place a staging environment at a later stage, our suggestion would be to implement staging and production from the start of the implementation.

Experiences EFI CW&D:

- The EFI over Websockets is stable and future proof.
- The XML structure is complex and there is a minimum of documentation
- The XSD is not self-explaining and there need to be an examples how to implement
- For future projects we would implement the complete message structure of the XSD. Currently we only use a part it.
- To make the EFI protocol future prove it needs to be in a different format (for example Json) because the XML is too heavy weight

2.5. Smart Storage Unit

The smart storage unit (SSU) installed at the Strijp-S site is a movable 20 feet reefer container with sound suppression. The container is fully insulated and is equipped with temperature and humidity control. The SSU consists of several units together such as battery cell storage racks, battery management system (BMS), two bi-directional inverters with LCL filters, energy management system (EMS), data acquisition system (DAS) and climate control equipment.

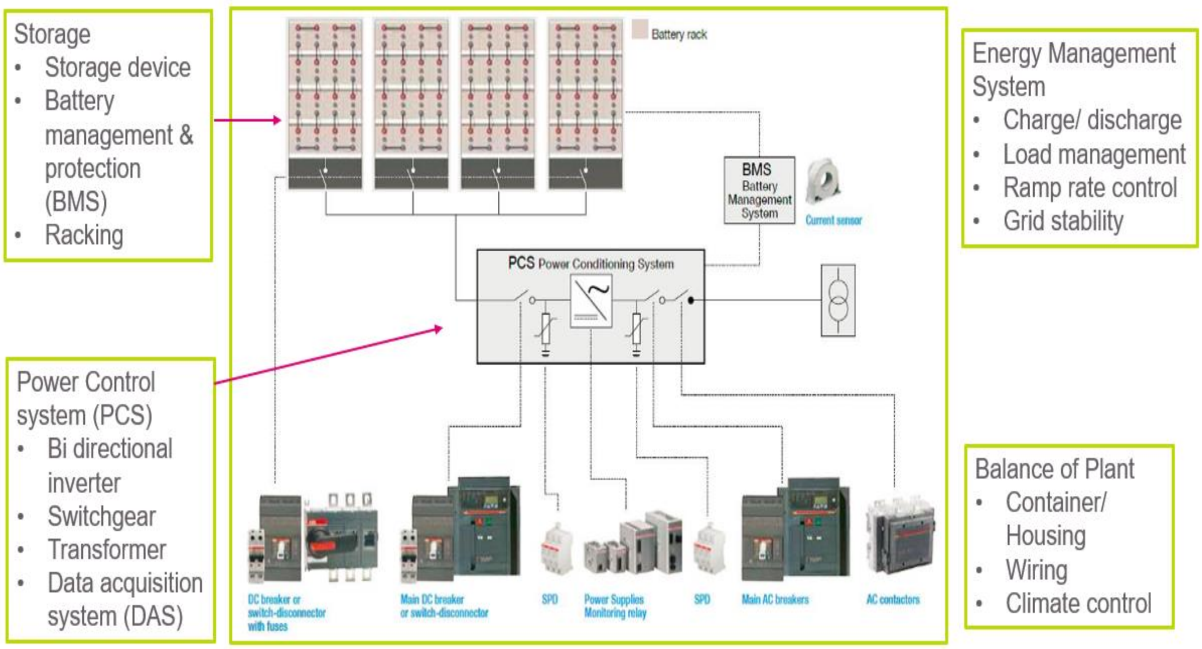

Figure 5 - Typical configuration of SSU

The battery storage uses Li-NMC (Lithium nickel manganese cobalt oxide) technology and has 126 modules. Each module is of 2,5 kW, 48 V. All the modules are arranged in 9 racks. Each rack consists of 14 battery modules and its own battery management system and fuses. Each rack is secured and is connected to a central busbar. The Master Controller collects the data and is the interfaces towards the Energy Trading control platform and/or the customer's own control platform.

The installed SSU capacity is 315 kWh, with converters capacity of 255 kW (bi-directional). The converter system consists of two separate inverters: one with a maximum power of 225 kW and the other one with 125 kW. This configuration allows to reduce the total system losses. Their working order is controlled by the EMS so that during periods of low demand only the small converter is used, while the larger one is only used when more than 125 kW is needed. The converter efficiency is 97% and the battery efficiency is 98.5%. Therefore, the overall system efficiency becomes 93% approximately. The SSU is being utilized mainly for demand side congestion management and some voltage support purposes. The LCL filters of the inverters are designed to provide certain amount of grid voltage support, including short-term power quality problems such as voltage dip, flicker suppression, and reactive power support.

For a battery, the charging and discharge currents are very important parameters as they determine the ageing rate and life expectancy of the battery system. The SSU of this pilot has a continuous discharge current of 75 A, while the charging current is 25 A. The state of charge (SoC) during storage and the temperature during operation also affect the ageing rate of the battery. The ageing rate of this type of Li-ion battery is around 0.5% of the total capacity after the first year and increases gradually to 0.85% in the 10th year. The SSU has a life expectancy of around 10 years.

Tender process

The SSU was purchased conform the applicable tender procedures allowing full transparency and equal treatment of qualified proposals according to pre-defined criteria laid out and published in the tender documents.
In total 4 ('four') proposals were withheld as being 'qualified' from the following bidders Alfen, Tesla, ATEPS and Eaton. These 4 offers were then carefully evaluated and rated upon 7 selection criteria each having a different importance and thus weighing factor to the overall rating. These criteria were:

1. Technical quality & compliance of deliverables (weighing factor of 40%)
2. Pricing & commercial offer (weighing factor of 35%)
3. Schedule & time-2-deliver (weighing factor of 10%)
4. Scope coverage (weighing factor of 5%)
5. Expertise & competences of supplier (weighing factor of 3%)
6. Warranties - to timely meet requirements (weighing factor of 3%)
7. Other advantages proposed (weighing factor of 4%)

Based upon these criteria, the proposal of ATEPS achieved the best overall score of 55 points (from a total of 100) and was awarded with the contract. This contract consisted of two key elements i.e. the purchase/supply of the SSU and the timely implementation a.k.a. installation at the project location. Since the project team had doubts whether certain project milestones would be met on time, an addendum to the standard contract with ATEPS was made, making sure that ATEPS would take all those actions required to meet the deadline of 05/09/18 of having completed a successful SAT test, performed by DNV GL.

Testing

Before the Factory Acceptance Test (FAT), scheduled in March 2018, could be done a problem was discovered with the net filter. This results in a 6 weeks drawback before a new net filter was available and installed from Finland. Meantime the time slot reserved for the FAT at the KEMA laboratory in Arnhem had been expired. The first possible free slot was at the end of August 2018.

Figure 6 - SSU inside the battery cabinet

The FAT started with a visual check in and around the system before the grid connection is put in place. Overall the system is according industry practice besides some minor comments on a faulty label a wrong identification on drawing. The issues were in the punch list for the SAT.
After grid connection the safety stop was successfully tested. An issue occurred with one of the breakers that resulted in the delay of the tests of one day, after solving tests resumed. The power step test, where the system had to operate on different power levels, resulted in the power levels not being according expectation. This was caused by the way of operating the system, a control loop was not in place, so power set point was derived on the inverters power estimation. This item was also put on punch list, during SAT. A working control algorithm must be in place to control power output at grid connection on a stable setpoint level.
The performance tests consisted of standby loss tests and two cycling tests. This resulted in a standby energy use of the system according expectation. With the circuit breakers closed the systems shows a large reactive component due the filters trying the filter the grid, this will in actual operation not take place because breakers will only be closed during charging or discharging.
Due to the not working of the power control loop, the power at both tests show a disturbed patron, and explanation was provided by the supplier of the SSU. The round-trip efficiency

was below the requirement of 93%. The test organization had recommended repeating the tests during operation to gain a realistic insight in the round-trip efficiency.
The total harmonic distortion during both tests was below the required value of 5%, which means the total harmonic distortion (THD) test passed criteria.
Finally, an extra dynamic test was performed which shows the system was capable of switching between full power charge and discharge in a short and acceptable time of 1.3 sec.

Installation

After the FAT the SSU was moved from Arnhem to Eindhoven.

Figure 7 - SSU at project site (outside battery cabinet)

All the batteries were removed as a safety measure, required by the RDW (the Dutch road traffic authority, from the container during transport and transported separately to Eindhoven.
After installing the batteries, a Site Acceptance Test (SAT) was performed on 5 September 2018. The items put on the FAT punch list should been handled by the manufacturer. It turned out that only one of the two inverters worked, the associated inverter controller not yet and that the power regulation based on the kWh meter didn't properly function.
The control circuit was oscillating. The communication to the LIMS was not working so all the tests could only been done locally. The LIMS communication could been tested with a delay of two weeks. During the SAT, various discharge / charging sessions were carried out in steps of 10 and 25 kW by means of a PQA measurement (Power Quality Analysis). The SSU responded well.

The value of the SSU was ultimately, software-wise, set to a power of 160 kW related to the maximum connected load of the connection.
The interface between the LIMS kept several problems that occurred 'spontaneous' the SSU still didn't function. This results in pointing form the SSU supplier to the LIMS supplier.

The SSU was delivered functioned well, except for the LIMS link on 19 September 2018. Initially, the controlling of the LIMS to the SSU seemed to be functioned well on 9 October 2018. However, due to software changes in the SSU, connection and controlling proved to be no longer possible. In the end, it took us until 7 December 2018 to drive the SSU from the LIMS. LIMS communication problems occurred again in March 2019. And until now (24th of Mai 2019) the system communication isn't stable witch influences the field test in a negative way. Enexis escalated this on board level with the supplier.

Simulation SSU

The simulation of the SSU is a software module in the LIMS. This way it is possible to switch between simulation and the actual SSU system during the project. Aggregators don't see any difference, except that the simulation will always act according to an ideal situation.

The simulation was also used as a way to test the architecture and communication of the LIMS system with the aggregators. The simulation can be enabled/disabled using the LIMS Web interface and either simulation or the actual SSU can be active, as they are using the same WebSocket connection.

The simulation provides the following parameters, as displayed in figure 8:

- Battery state of charge [%]
- Power requested [W]
- Power delivered [W]
- Mode of operation [charging, discharging, idle]
- Container temperature [°C]
- 3-phase voltage [V]
- HVAC status [on/off]

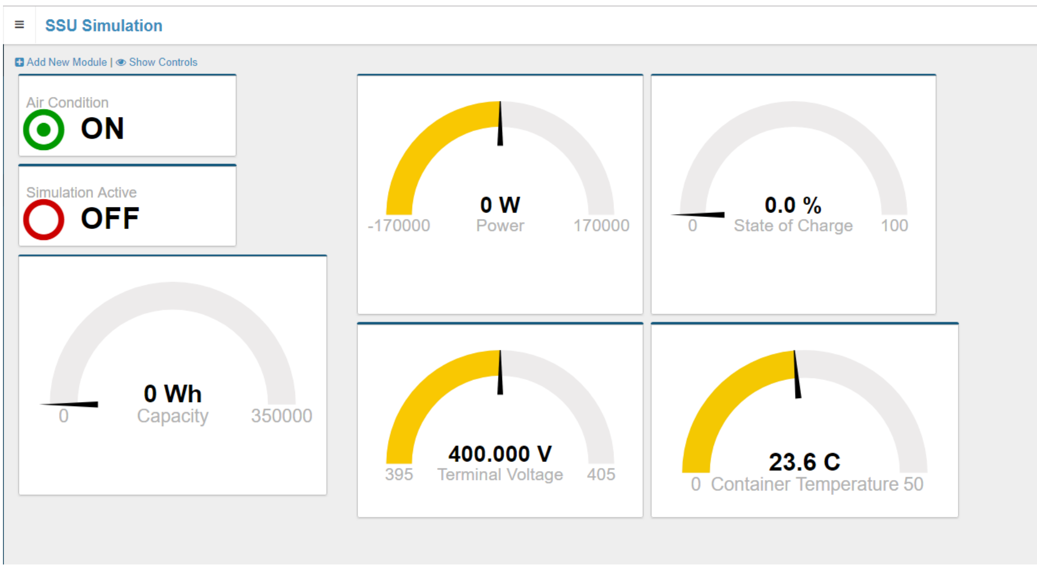

Figure 8 - Simulation parameters

The simulation has an interface to change simulated battery characteristics:

- Capacity [Wh]
- Max/min power [W]
- Efficiency [0-1]
- Losses [W]

Experiences SSU:

The battery business is relatively new on the DSO market level. This results in new players that offers out of the box solutions. Adapting them to specific needs is difficult for them. Experience and lack of manpower makes it hard to react on problems in a proper way.

- The option of having a simulation as a fall-back scenario makes the pilot testing more robust. It allows smooth progress in chain testing also in situations when the real SSU system is in maintenance or worse, defective. Given that the connected SSU is in its technical infancy this proved to be a valuable part of software.
- Since simulation data is not real data, it may not always accurately reflect the reality. Practical problems are less likely to occur in a simulation data, possibly resulting in a too positive perspective of the systems performance.
- The SSU was also in development phase, this gave some interfacing problems because there was no documentation on control and monitoring of the SSU.
- The SSU interface needs to give more information about the actual status of the SSU and the related parameters when the SSU is in error state.
- The boundaries of the loading curve (charging – discharging) are not fixed and need to be also accessible in the interface software.

2.6. PV panels

The connected PV panels are “rotating PV”. This is a concept developed by the company Softs and marketed as “Softs-spot”. See the image below:

Figure 9 - Left actual PV system controlled, right Softs logo

It consists of 6 motorised arrays of 9 PV panels each. The PV panels are on a triangular frame as indicated in the image below and can be controlled per array to be put in one of 3 positions:

- PV
- Advertisement
- Filter against air pollution.

Figure 10 - Rotating PV triangular frame

The rotating makes the PV output adjustable. For the sake of InterFlex we in practice only switch between two positions:

- PV
- Advertisement

Simulation PV

The simulation of the PV is a software module in the LIMS. This way it is possible to switch between simulation and the actual PV system during the project. Aggregators don't see any difference, except that the simulation forecast will have a high degree of accuracy. The simulation is based on the measurements from the weather station and solar radiance forecasting/measurements.

The simulation PV can be curtailed in steps of 10%, thus giving higher granularity in control than for the actual PV system. Additionally, curtailment is always successful i.e. curtailment of 50% will result in exactly 50% lower power than forecasted.
The simulation can be enabled/disabled using the LIMS Web interface. Either simulation can be active or the actual PV, as they are using the same WebSocket connection.

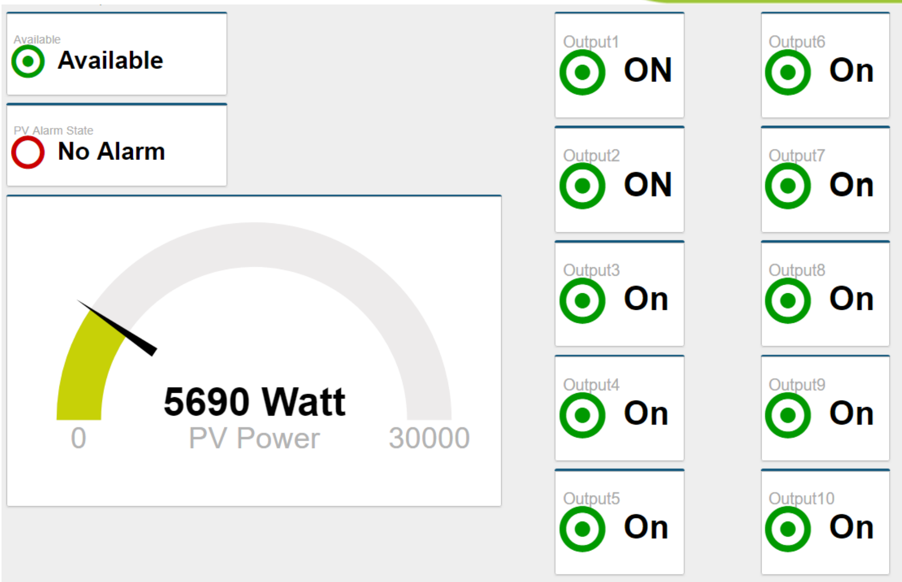

Figure 11 – PV simulation display

Experiences PV panels

- The solution to interface via proven protocols proved itself. It works reliably with no trouble.
- The option of having a simulation as a fall back scenario makes the pilot testing more robust. It allows smooth progress in chain testing also in situations when the real system is in maintenance or worse, defective. Given that the connected rotating PV is in its technical infancy this proved to be a valuable part of software.
- The rotating PV technology is in its infancy: For example, PV rotate to undefined states or fail to rotate. The calibration of the encoders is also not ideal and panels can end up in a half solar, half advertisement position, resulting in a system unknown state.
- The measurement points in the system are too little to really suit LIMS needs. There is only one measurement for the energy of the entire PV system.
- The power measurements are read through Modbus from the PLC that is controlling the panels. As such, LIMS cannot know if the power reading supplied by the PLC is correct, it can only read the values provided. In case of software misconfiguration of the PLC, the LIMS would use incorrect values.

2.7. Solar Car

Stella Vie is a solar-powered family car built by Solar Team Eindhoven, which consists of students of the Eindhoven University of Technology. The goal of the car is to demonstrate that it is possible to drive on solar power in a normal way. Next to that, Stella Vie and her team won the Bridgestone World Solar Challenge 2017 in Australia, world’s largest competition for solar cars.

Figure 12 - Solar Car at Interflex meeting in Eindhoven

Stella Vie can seat 5 persons and is fully road legal in Europe. She has two highly efficient in-wheel motors and a top speed of 125 km/h. On the battery of 12,5 kWh, she can drive 600 to 700 km. The roof is covered with 326 solar cells that can deliver 1,3 kWp. The range of the car can therefore be extended up to a 1000 km with energy produced by the solar cells. Next to this, the team developed several concepts that help the driver to drive as efficiently as possible. An example is the solar navigator and parking system. This system helps the driver to pick the most efficient route based on weather data and to park the car in the sun as long as possible based on data of the position of the sun and the built environment. Besides this, a driving assistance system can give feedback on the driving style in terms of colour signals (red to green).

One of the most interesting features is the energy management system. A solar car can be seen as a mobile solar panel that can generate energy wherever you want. To make optimal use of this energy, the user should be able to share a surplus of energy to the grid to, for example, power his house.

Charging station

To make use of the energy management system, it is required to use a bidirectional charger that cannot only charge a car battery but also discharge a car battery. In search of such a bidirectional charger, the team found out that none bidirectional chargers were available.

Therefore, a charging station is currently under development to enable charging and discharging and to use the energy in the Strijp-S demonstration.
The charger is built of existing hardware of industry partners and newly developed hard- & software. With a simple user interface, the user can decide itself if he wants to charge the car, to discharge the car or to let the software system decide.

The Energy Resource Manager software of CroonWolter&Dros can decide - based on the forecast algorithm - to charge or discharge the car depending on the needs of the customer (driver) in combination with the needs of the commercial aggregator. The bidirectional functionality of Stella Vie gives the Energy Resource Manager the maximum flexibility to control the balance of the net on a specific congestion point where the charging station and Stella Vie vehicle is situated.

2.8. CroonWolter&Dros building.

As an extra the CroonWolter&Dros office building at the Limburglaan 38 in Eindhoven is connected to the LIMS through the office meter interface (DLMS) and connected to secured, internal network of CroonWolter&Dros.

Figure 13 – Connected office building

The meter provides information about the energy usage via 2 counters, for the day tariff and night tariff. Based on the current counter status [kWh] and last counter status 15 min usage is calculated.

The consumption data can be seen on the LIMS web interface. Data is shown as kWh/15 min.

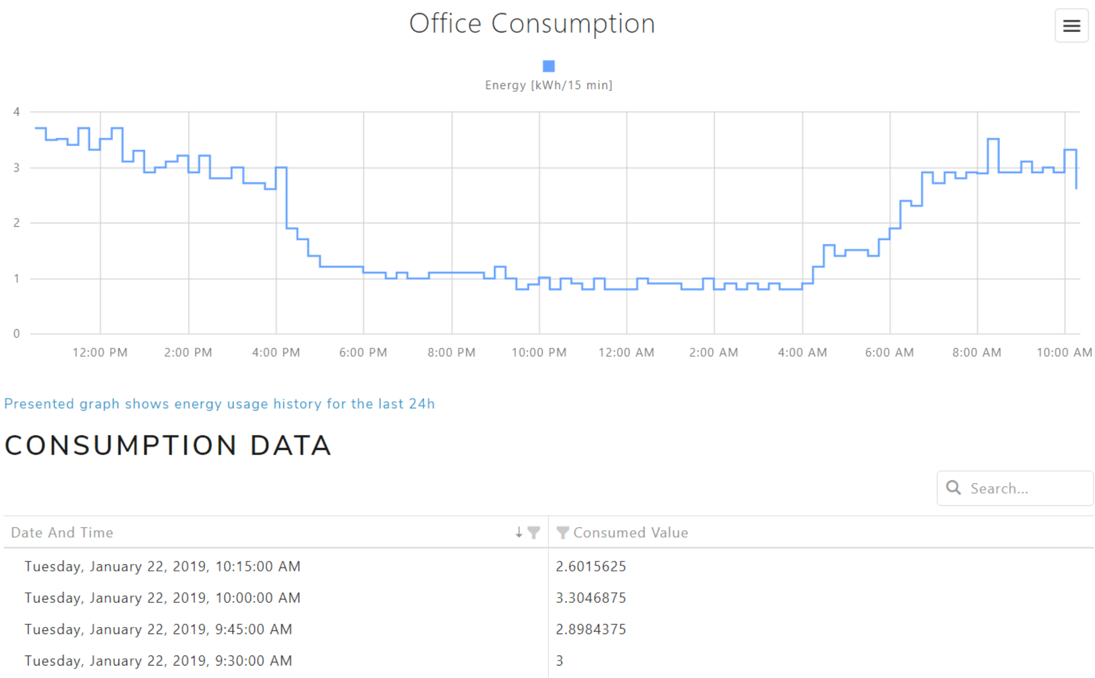

Figure 14 – Consumption data in the web interface

3. CHAIN TEST

This chapter describes the chain test.

Chain test principle

The purpose of the chain test was to validate whether the systems (see figure 1) of the flexibility market process perform from end to end according to the specified requirements. We tested happy and unhappy flows. In a happy flow scenario messages have been handled in line with the agreed process; messages in the chain are sent as expected. In a unhappy scenario the software chain is somewhere interrupted. We made a distinction between technical problems (f.e. system failure, malfunction of devices) and functional problems (f.e. not all flex delivered). For both scenarios, happy and unhappy flows, it must be clear how they should be handled (f.e. sending an error message in case of disruption).

In a sprint backlog of the chain test we kept track of all the test scenarios and the issues to be solved. The flow steps and status were described in a test result document. A few times a week the progress was discussed with the test coordinators in a stand-up meeting project

Chain test experiences use case 1

In the requirements it was not defined that more than one commercial aggregator can control the SSU (or PV) at the same time. For the SSU we could have stated that each commercial aggregator could control a part of the total capacity, e.g. each 50 percent. The question is of course whether that is a realistic scenario. Since we did not split capacity, we needed double time to test both DER (SSU and PV) with both commercial aggregators (Sympower and TNO). When Sympower tested the SSU, TNO tested the PV and vice versa.

Chain test experiences test use case 1:

- The SSU and PV were not active (and connected to the LIMS) on time, therefore the chain tests were started with dummy measurements and status provided by the LIMS. We started with testing the happy flow message chain, with minimal focus on content.
- Unfortunately, the SSU and PV were only ready for testing after the 6 week chain test. Then, of course the LIMS communication test with these DER was performed first. Followed by a ‘real’ chain test per commercial aggregator, with unfortunately limited historical data.
- The rotating PV we use in this project is too small to have impact on the flex market. Lesson learned, if the research analysts were involved earlier, we could have introduced the PV factor earlier.
- For some partners the implementation was not completely finished before the chain test started. Some part of handling the EFI + protocol, concerning the instruction to the SSU and PV, had to be built during chain test. Also, all partners were still optimizing part of their core business rules.
- Also, discussions were started about the content of some messages. It seemed that during implementation phase not all detailed requirements were clear or not looked into yet. For example:
    - positive or negative values (production vs. consumption) in the different messages

    - influence summer/wintertime
- Finally, what must be taken into account with the chaintest experience above, none of the project parties were fulltime available for specification, implementation and chaintest phase. Furthermore, there were absences due to sickness and holiday. Looking back, with limited resources a lot of work was accomplished.
- Due to the late requirement discussions with regard to flex settlement, the field test starts with some open items, which were solved during the first two weeks of the tests.
- The systems that were built from scratch for InterFlex, do not have a test environment. For testing extra functionalities this is needed to prevent disturbance of the chain test.
- LIMS provided simulation measurements and forecasts for SSU and PV, that made it possible to start test the EFI + message flows. Simulation must best be available before you start the chaintest.
- The EFI + protocol could be tested completely but not the actual messaging to the SSU and PV since they were not active. We needed an extra chaintest for that part of the chain.
- During the chaintest many system issues were found and most were solved. The stand-up meetings with test coordinators were very helpful to keep focus.
- Activate the SSU and PV in the chain was not as easy as thought. The SSU still is not working properly after the chaintest. The initial chaintest was done without real connection to SSU and PV.
- During chaintest it became clear that some essential functionalities were not defined completely during implementation phase; this delayed the chaintest.
- With respect to privacy, the most important aspect to consider is whether personal information is being exchanged. In the case of this project, there is no exchange of personal information. However, there is a possibility that energy consumption data and corresponding EAN-numbers are being published. If this is the case, exchange of personal information might occur and the corresponding privacy regulations need to be dealt with accordingly.
- In future projects where the LIMS is being used the software would be integrated in the network of the customer and would only be accessible within the company network whereby the company privacy policy would be applicable.
- The systems and protocols built for the Interflex project are built by 6 parties. This makes it a challenge to coordinate and plan the different project phases and deliverables. The project is managed on a partly traditional project management concept based on the waterfall methodology in combination with Agile/scrum. This has a big impact on the resources planning.

4. APPROACH AND RESULTS SCENARIOS

This chapter describes the research approach and the preliminary results.

4.1. Pilot set up

In this project, the day-ahead load forecast is run for two MV/LV transformers. See fig. 15 for the congestion points topology and specifications. Forecasting the inflexible load is done separately from flexible load. In the rest of the report, inflexible load forecast will be addressed as “Load Forecast”, and flexible load forecast (e.g. PV, SSU, EV) will be referred to as “D-prognosis”. Fig. 16 and Fig. 17 demonstrate the day-ahead load forecast of two congestion points during one week of April as an example. In these figures, the red curve shows the real measurements of the grid and the blue curve represents the load forecast. For both measurements and the forecast, the flexible load feeder is excluded, since the flexible and inflexible loads are predicted separately. On one side, the inflexible load based on the DALi measurements will be forecasted by Enexis. On the other side, each aggregator is responsible of forecasting the flexible load which is under their operation. The day-ahead flexible load forecast which is called D-prognosis will be sent by aggregators to the Grid Management System (GMS) in order to be added to the inflexible load forecast. Finally, the total load profile as an input for the Flex Decision Module (FDM) can be created.

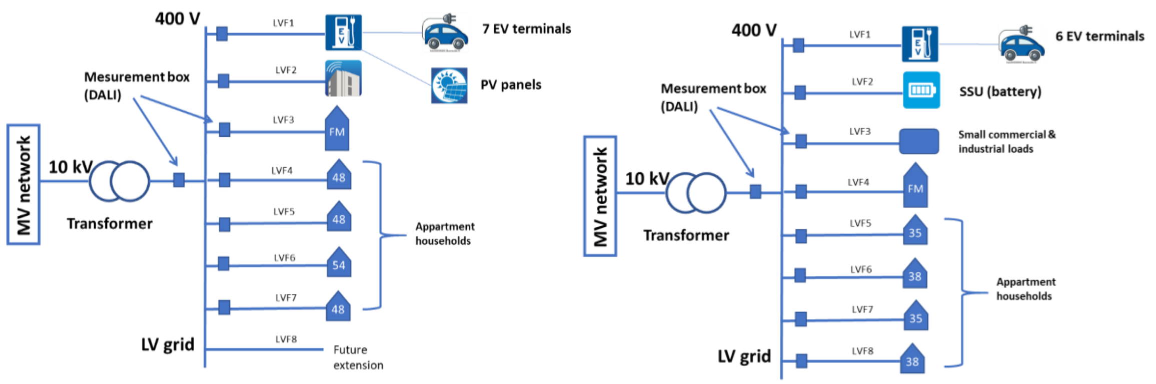

Figure 15 - Congestion points topology and specifications

Flex Decision module will determine the day-ahead Flex Need based on the total load forecast and the Demo Max. In case that peak load exceeds the Demo Max, a flex need will be generated. It considers both positive and negative Demo Max thresholds i.e. the possibility of reverse power flow due to high PV generation. Flex need includes the magnitude of required flexibility, as well as the maximum desirable price and sanction price for each 96 PTU’s (Programme Time Unit) in a day, and each congestion point individually. A flex need will be sent as a Flex Request to each aggregator connected to each congestion point under two conditions. First, D-prognosis has been received from the aggregator; second, aggregator responds within a specific time window that the USEF message has been received by them. If the aggregator does not send any D-prognosis or if it fails to receive the message, there will be no Flex Request, even though there is a flex need. However, the initial analysis shows that in more than 90% of the time, a flex need leads to a flex request. In case of overloading, flex request has negative sign which means the load must be reduced.

In case of overproduction, flex request is positive which implies the load must be increased, or local energy generation be reduced.

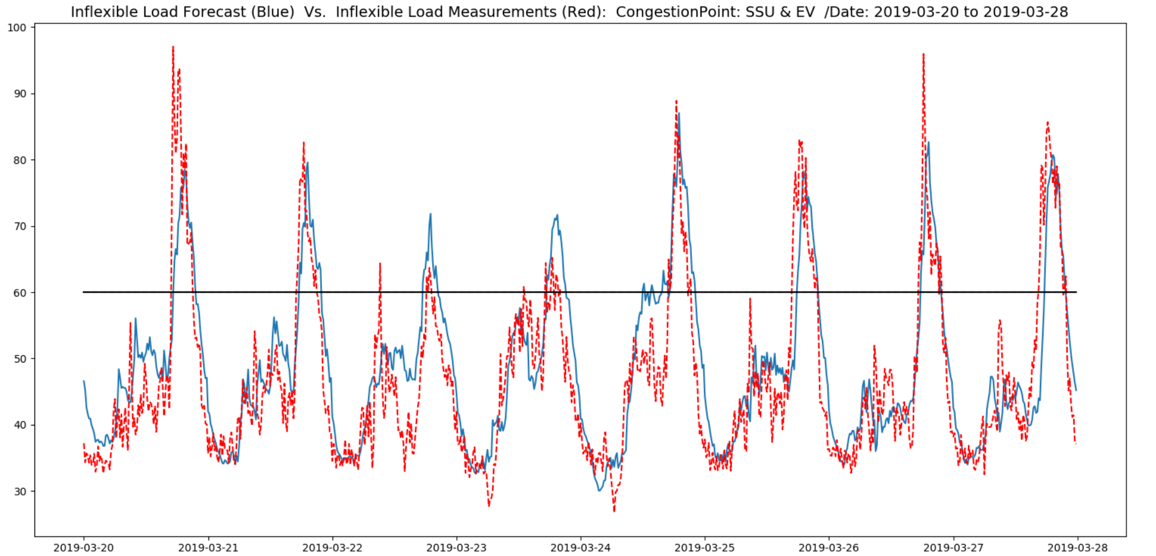

Figure 16 - Congestion point 5 (SSU & EV street)

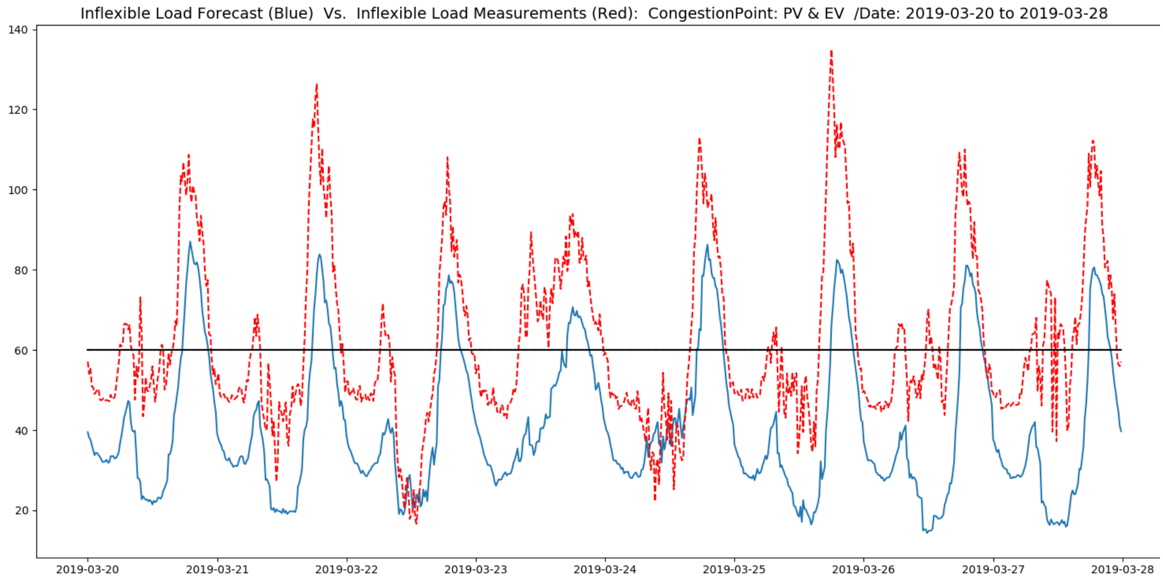

Figure 17 - Congestion point 8 (PV & EV parking)

4.2. Goal field tests

During the field tests the developed systems will be tested in a real life setting. Goal of the field tests is to assess the operation of a local flexibility market under realistic operating conditions and identify criteria for the proper functioning such a market. During the field test different market scenarios will be tested under which the flexibility is procured.

Test scenarios

Scenario 1: Day-ahead market
In this phase all flexibility in the local market is traded via a day-ahead market. Based on day-ahead forecasts the flexibility is requested, offered and (if possible) traded. Subsequently the different parties will follow the agreed upon profile for the next day. During the day no more deviations are possible on the local market.

Scenario 1a: day-ahead market with Sympower
In this scenario commercial aggregator Sympower controls the SSU and the PV; and Jedlix controls the EV.

Scenario 1b: day-ahead market with TNO
In this scenario commercial aggregatorTNO controls the SSU and the PV; and Jedlix controls the EV.

4.3. Research approach

In this section the different research questions are outlined. Subsequently, sub questions are formulated per research question, and details are given on how these questions will be answered in the research. The research questions are formulated as follows:

1. How much flexible power and energy is needed in the network? When? Where?
2. How much flexible power is available in the network? When? Where?
3. What is the realized flexible power in the area?

Figure 18 provides an overview of the relations between research questions 1 till 3. Question 1 focuses on the flex requested in the market. Question 2 focuses on what flex is offered in the market, where an overlap arises with the requests. This overlap is analysed in question 3.

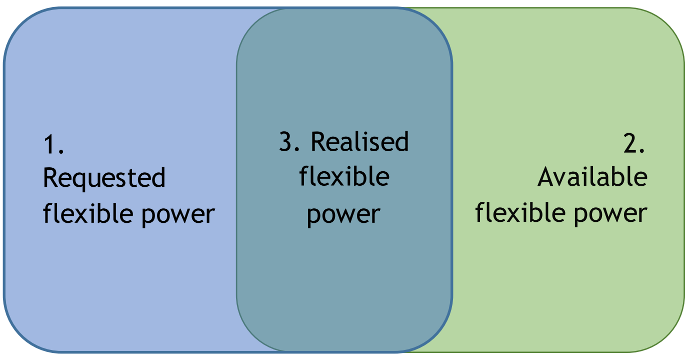

Figure 18 - Relation between research questions 1-3

Research question 1 – Necessary flexible power and energy

Sub questions

a. At what moments, what level of flexible power is requested?

b. At which congestion points, what level of flexible power is requested?
c. How much flexible energy is requested over the course of the day and other testing periods?

Data and methods

Based on the requested output profiles of the flex decision module (FlexDecision) it will be analysed how much flexible power was requested at each moment in time and at which congestion points this was needed. Additionally these profiles will be analysed to assess the total amount of energy requested over different periods (e.g. daily, weekly). The first analysis will focus more on the detailed level of the need for flexibility in numbers and exact location, whereas the second analysis will yield a more cumulative insight over time.

Analyses:

Plots will be made per congestion point for the flexibility profiles, the counts of upward or downward flexibility requested at each PTU and a distribution of the requested flexible power.

Research question 2 – Available flexible power

Sub questions

a. At what times, what level of flexible power is offered?
b. At which congestion points, what level of flexible power is offered?
c. Who offers the flexible power?
d. How much flexible energy is being offered during the day and over different testing periods?

Data and methods

Based on the offered flexibility profiles (FlexOffer) it will be analysed how much flexible power was offered at each moment in time and at which congestion points. Additionally these profiles will be analysed to assess the total amount of energy offered over different periods (e.g. daily, weekly). The first analysis will focus more on the detailed level of the offered for flexibility in numbers and exact location, whereas the second analysis will yield a more cumulative insight over time.

Analysis:

Plots will be made per congestion point for the flexibility profiles, the counts of upward or downward flexibility offered at each PTU and a distribution of the offered flexible power, also relative to the amount of requested flexible power.

The following KPIs will be used to assess the offered flexibility.

The relative amount of flexibility made available by the LA to the CA:

$$\text{Flexibility}_{\%\,available\,field} = \frac{\sum P_{flexibility\,offered\,by\,LA\,to\,CA}}{\sum P_{Total\,theoretically\,available\,flexibility}} * 100$$

Research question 3 – Realised flexible power

Sub questions

a. At what moments, what level of flexible power is realised?

b. At which congestion points, what level of flexible power is realised?
c. How much flexible energy is exchanged over the course of different time periods (e.g. daily, weekly, …)
d. At how many and which moments does the offered flexibility (not) match the requested flexibility?
e. What is the ratio between the actual realised flexibility and the traded flexibility?
f. What is the ratio between the traded flexibility and the forecast available power?

Data and methods
Based on the actual traded and realised flexibility profiles (FlexOrder, FlexOrderSettlement) it will be analysed how much flexible power was traded and at which congestion point. Additionally an analysis will be made on these profiles to assess the total amount of traded flexible energy. Furthermore a comparison will be made with the actual measured and realised profiles. Again insight will be made on a detailed level per PTU and per congestion point, as well as a more cumulative insight over several time periods. A more detailed analysis will also be done on the flex requests which could not be met (see Figure 2) and why.

The following KPIs will be used to assess the traded and realised flexibility:
The amount of flexibility traded with the DSO relative to the amount offered by the CA on the local market:

$$\text{Flexibility}_{\%\ traded\ DSO} = \frac{\sum P_{flexibility\ traded\ with\ DSO}}{\sum P_{flexibility\ employed\ on\ markets\ by\ CA}} * 100$$

The amount of flexibility offered on the local market by the CA relative to the amount it could theoretically offer:

$$\text{Flexibility}_{\%\ traded\ markets} = \frac{\sum P_{flexibility\ employed\ on\ markets\ by\ CA}}{\sum P_{flexibility\ offered\ by\ LA\ to\ CA}} * 100$$

The amount of flexibility traded with the DSO relative to the amount requested by the DSO:

$$\text{Flexibility}_{\%\ obtained\ DSO} = \frac{\sum P_{flexibility\ traded\ with\ DSO}}{\sum P_{flexibility\ requested\ by\ DSO}} * 100$$

4.4. Preliminary results

Setup congestion point 5

SSU is connected to one congestion point together with EV street, and the PV is connected to another congestion point together with EV parking garage. The Flex Request is sent to all the aggregators connected to each congestion point for the next 96 PTUs (every 15 min) in the day ahead. This message includes the magnitude of the required flexibility (kW), the positive or negative sign (increasing/decreasing the load), the max price and the sanction price per PTU. In response to this message, each aggregator will send a Flex Offer which also includes the same information.

Load Forecast

As you can observe in the Fig. 16 and 17, the load forecast of low voltage distribution grid and individual transformers cannot be as accurate as forecasting the aggregated load profile in the medium voltage level. Therefore, there is a noticeable forecast error which is due to fluctuations of low voltage load and can lead to high uncertainty. Aside from inflexible load forecast, D-prognosis done by aggregators for flexible loads such as EV, PV and SSU will also add to the forecast error. In the case of flexible loads, volatility and consequently, the uncertainty is even higher.
For both points, the congestion has been created virtually, since the real capacity of the transformer is quite higher than the peak load. In both figures, the black line illustrates the Demonstration Maximum (Demo Max) capacity of the transformer to cause virtual congestion. Demo Max value affect the magnitude and the occurrence of the congestion; hence, determining this factor has strong impact on the research results, particularly the business model of the flex market. Moreover, it is important to keep the Demo Max value fixed during the period of each scenario; otherwise, it can lead to deceptive results. At the moment, the Demo Max for both transformers is set on 50 kW; however, it is still under experiment and expected to be adjusted again.

Flex Request & Flex Offer

Fig. 19 illustrates the comparison between Flex Request and Flex Offer for SSU, and fig. 20 shows the Flex Order. A Flex Request from the DSO can result in a Flex Offer to the aggregator. The Flex Offer can result in Flex order from the DSO to the aggregator. Fig. 21 shows the Flex Request for PV. The figures show the average value for the whole period of 1st till end of April. In this period of time, all the requested power values are negative which implies the overloading status. As it can be observed in Fig. 19, SSU can comply with the requested power very well. Flex offer deviation from flex request is quite negligible. Therefore, SSU as a flexibility resource can provide high availability to solve congestion problems. On the other hand, there is no flex offer from PV side yet, which is obviously due to sending negative flex request. In this pilot project, there are 6 columns with each 9 PV panels. Hence, the PV curtailment can be conducted in 6 steps.

Flex Order & Flex Supply

Fig. 6 demonstrates the Flex Order for SSU. As it can be observed, flex order conforms to flex offer from both timing and value point of view. According to these results, flex offer must have been in the desirable price margin of Enexis; therefore, it has been led to an order. Nonetheless, there is no flex supply during this period of time which was due to the maintenance of the battery storage; hence, the ordered flex could not be delivered. Due to this issue, it is not possible to make any solid conclusion regarding the reliability of this flexibility resource yet.

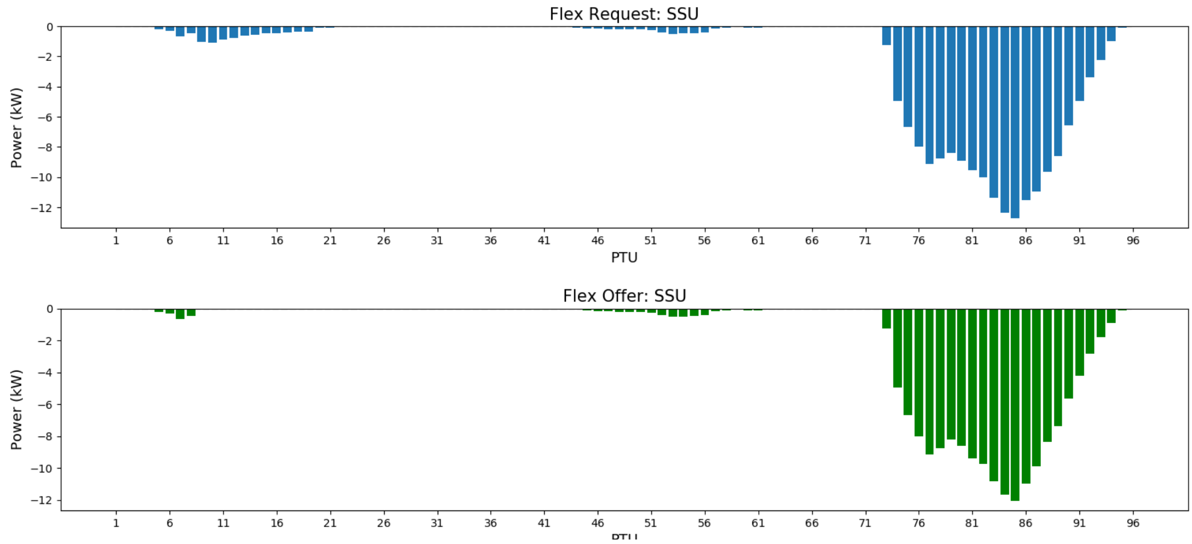

Figure 19 - Flex Request Vs. Flex Offer: Average per PTU

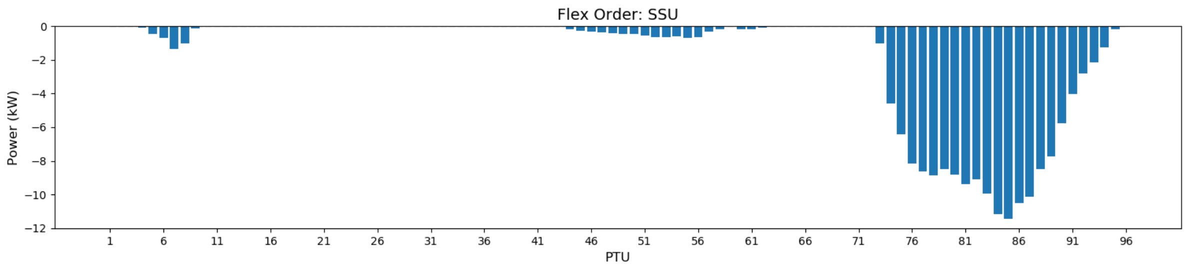

Figure 20 - Flex Order: Average per PTU

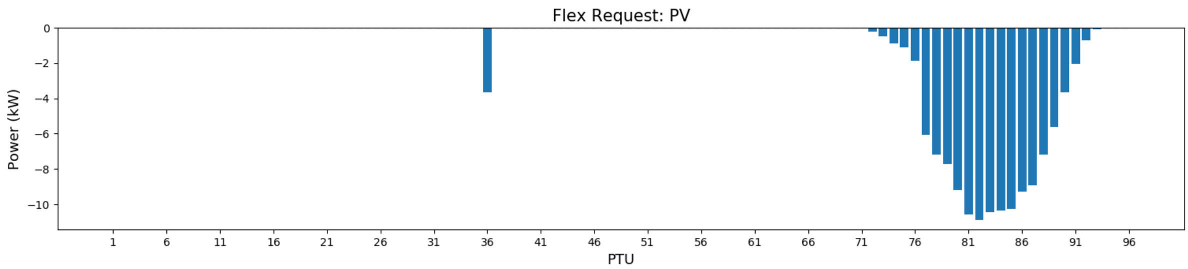

Figure 21 - Flex Request: Average per PTU

5. CONCLUSIONS & RECOMMENDATIONS

In this chapter describes the conclusions and recommendations regarding use case 2 and 3. General lessons learned that apply for all three use cases, will be input for deliverable 7.7.

LIMS:

Simulations are very usefull.
In order to the complexity and the dependency of the different DER it was a wise decision to design simulation software for chain testing and interoperability testing between the different systems. Also the simulation are partly been used for the field testing.

SSU:

The battery is not used on the TSO market.
The battery is to small for deployment on the TSO market by the aggargators and to pool with other assets.

Battery is to 'big' for the DSO market
The sizing of the battery is depending on the needs. The needs on the DSO market could better be served with a few smaller batteries then with one large one. This makes it also possible for an aggregator to put his batteries on different markets.

Battery specifications to complex.
The battery was intended to solve congestion and power quality issues. There for the specifications of the battery and specific the converters was complex. This reduces the amount of potential suppliers or introduces suppliers who think they can fix it when needed.

Market tender.
The key lessons learned is that a more detailed reference-check of the bidder is necessary with respect to their performance in similar projects and the expertise-level of the project management in meeting project deadlines without delay. In that respect, the choice for a fairly new player was risky.
Other lessons learned are:
(i) to pre-define and stipulate contractual terms and conditions as much as possible upfront in the process;
(ii) demand acceptance thereof by the bidders as a condition for proceeding with the tender process; whilst (iii) ensuring 5-10% customizing these conditions at the award-stage of the process.

Roating PV, solar car and the CW&D building have no specific conclusions other that they were connected to the LIMs and worked as expected.

EFI:

Usability of the protocol.
EFI seems to be a good fit for the requirements of the InterFlex project. Although there is a learning curve when implementing EFI, all relevant partners were able to successful implement EFI.

Chain test:

It might have been better to wait with the chaintest until the SSU and PV were ready. But, we started the chaintest to keep focus and pressure. And a lot could be tested without the real SSU and PV connection, but it would have been easier with them active.

Preliminary results

In the field test the SSU was able to offer the requested flexible power quite accurately. Unfortunately, due to battery maintenance the SSU was not able to actually deliver the ordered flex on a continues bases.

In contrast, the requested flexibility did not match the flex capabilities of the PV system. The PV is only capable of sending flex offer in case of a positive flex request, which indicates an overproduction. By moving towards summer time, it is expected that PV generation will increase and cause reverse power flow. The effect is that in case of exceeding the negative Demo Max, the positive flex request for PV curtailment will be sent, and consequently a positive flex offer from PV will be expected.
Due to the complexity of the field test, so far no meaningful preliminary results could be obtained regarding the KPI's. There have been many changes, while fine tuning parameters

Recommendations:

The complexity in integrating different systems with different protocols and interfaces was and is the big challenge for this use case. Good and basic specification are the key words. For the LIMS connection with EFI this works fine. The protocol proves his usability in this. The complexity of the battery is the weak spot. The battery in this project isn't able to meet the expectations yet and by specifying the battery it should be clear what purpose the battery should serve in the network. Make it not to complex.

Overall testing between several partners mean also a lot of coordination. The project method of Agile used here proofs to be a good one. But this also means a lot of discipline by the different partners.

6. APPENDICES

6.1. Appendix 1: Proposal additions EFI +

Communication and Security

For the communication WebSockets is used. The CEM will function as a WebSocket Server, while the LIMS will function as a WebSocket client.
The WebSocket server will use a secured (HTTPS/WSS) connection with a (signed) certificate.
Authentication of the client can be solved in several ways:

- IP Whitelisting: Configure a firewall to only allow incoming connections from certain IP addresses
- Basic authentication: Using HTTP basic authentication a Websocket Client can provide a username and password
- Client certificates: The client use a (signed) certificate

Assets.

General

The general functionality of EFI + are:

Required information or functionality	Description	Proposal EFI +	Status
EAN Code	The identifier for the connection (in USEF terms)	Add a field to the abstract Registration message.	OK
Device ID	Identifier for the DER	Use the ResourceID in EFI.	OK
Congestion Point ID	Identifier for the congestion point (in USEF terms)	Don't add to EFI +. Congestion Points are defined by DSO, the DER has no use for this information. Based on the EAN code the AGR can query the Congestion Point ID from the CRO.	OK
Maintenance profile	A profile in which maintenance periods (where the device is not available) is communicated	Create a new message type for this purpose. LIMS pushes a new, complete, maintenance profile after sending a Registration or when new information is available. The new message contains a list of intervals in which the DER is not available. Interval has a startTime and a endTime, and optionally a reason.	OK

Mode (maintenance)	Indicates if the device is in normal (flexible) mode or unavailable (maintained mode).	The LIMS revokes flexibility using the FlexibilityRevoke whenever the device goes into maintenance. After maintenance new FlexibilityUpdate messages are sent, indicating flexibility to the CEM. We add an optional 'reason' field to indicate the reason for the flexibility revoke (maintenance, localcontrol or alarm). Additionally, when the LIMS is not available the FAP knows immediately since the WebSocket connection is lost (the FAP can make use of the ping feature of WebSockets).	OK
Alarm state	See if the device is unavailable due to unforeseen circumstances.	See Mode.	OK
Voltage	Nominal voltage	Don't add to EFI +. Voltage is not relevant for the AGR, only power and energy (for which the Measurement message can be used).	OK
Current	Current measurements	Don't add to EFI +. Current is not relevant for the AGR, only power and energy (for which the Measurement message can be used).	OK
Timestamp	Format used for time notations, used in several places	Use XML timestamp WITH timezone notation (e.g. use 'Z' for UTC). For example: 2002-05-30T09:30:10Z. In this project we will use UTC.	OK
MessageVersion	Version of the message, in cases where a message can be sent multiple times (with different content)	EFI uses unique messages, don't implement this	OK
MessageId	Unique identifier for each message	Use the existing specific identifiers (e.g. flexiblityUpdateId, instructionId…) in combination with timestamps. If we discover that this is not sufficient, we will implement MessageId later on	OK

96 PTU program / Base_Line	Desired power level for each PTU	**For storage**: Use a sequence of ActuatorInstructions with different startTimes. This can be done within one or multiple StorageInstruction objects. **For Inflexible**: Send a curtailment profile using the InflexibleInstruction object **Rationale:** EFI tries to make the least amount of assumptions about the working of the DER and the aggregator. It tries to hide all those specific details, in order to achieve maximum interoperability. In this project we use a market with 96 PTU's. That is a technical property of the aggregators used in Interflex, but there are also use cases where this isn't the case. The design philosophy of EFI dictates that this technical property should be hidden from the DER.	OK
Update_Base_Line	Change earlier given instructions	Use InstructionRevoke messages to revoke previous Instructions.	OK

Office Building Strijp-S

EFI Category
The device is classified as an Inflexible device.

Smart Storage Unit

EFI Category
Smart Storage Unit is defined as the Storage EFI Category.

Proposals

Required information or functionality	**Description**	**Proposal EFI +**	**Status**
Capacity usable	Available capacity of the battery that can be used for flexibility at which point in time	Define as the LowerBound and UpperBound of the fillLevel of the RuninngModes. We a new field 'validUntil' in the StorageSystemDescription (in addition to validFrom). This way the LIMS can define multiple systemdescriptions in different points in time. Two systemdescriptions are not allowed to overlap in time. The fillLevel might change if the system description changes. This is	OK

		communicated using the StorageStateUpdate message when the capacity changes; it is not communicated beforehand.	
Charging power	Available rate of charging	Define using RunningModes.	OK
Energy	Current state of charge	Define as fillLevel.	OK

PV roof

EFI Category
The device is classified as an Adjustable device. In my opinion, the Inflexible category makes more sense here. The Inflexible category supports curtailment.

Proposals

Required information or functionality	**Description**	**Proposal EFI +**	**Status**
Delta steps	Steps in which PV production can be reduced (percentage, no absolute values)	Define the steps in the CurtailmentOptions message. We add functionality to the CurtailmentOptions to either define options in absolute values (currently implemented in EFI) as percentages (newly implemented in EFI+)	OK
Angle	Angle between ? (North? Ground?) and PV panel.	Ignore. AGR knows this is a PV panel (part of Registration message), patterns can be detected from measurements. The LIMS can come up with a forecast. In situations where the LIMS cannot do this, a FAP could reverse engineer the angle by looking at measurement data.	OK
Nominal power	Maximal production and consumption power values for this DER	Add new required fields in the InflexibleRegistration for maximal production (positive, in Watt) and maximal consumption to the abstract Registration message.	OK

6.2. Appendix 2: Specification EFI + over WebSockets

Introduction
This appendix describes how WebSockets is used for EFI+ communication.
EFI+ has four categories (Inflexible, Shiftable, Adjustable and Storage), with each their own set of messages. A DER uses only one of the four categories.
For the EFI+ WebSocket communication, the LIMS (ResourceManager in EFI terms) takes the role of the WebSocket client, while the FAP (Customer Energy Manager in EFI terms) takes the role of the WebSocket server. DER connects to only one FAP, but the FAP can connect to many DERs. The LIMS can represent multiple DERs, so there can be multiple simultaneous WebSocket Sessions between the LIMS and the FAP.

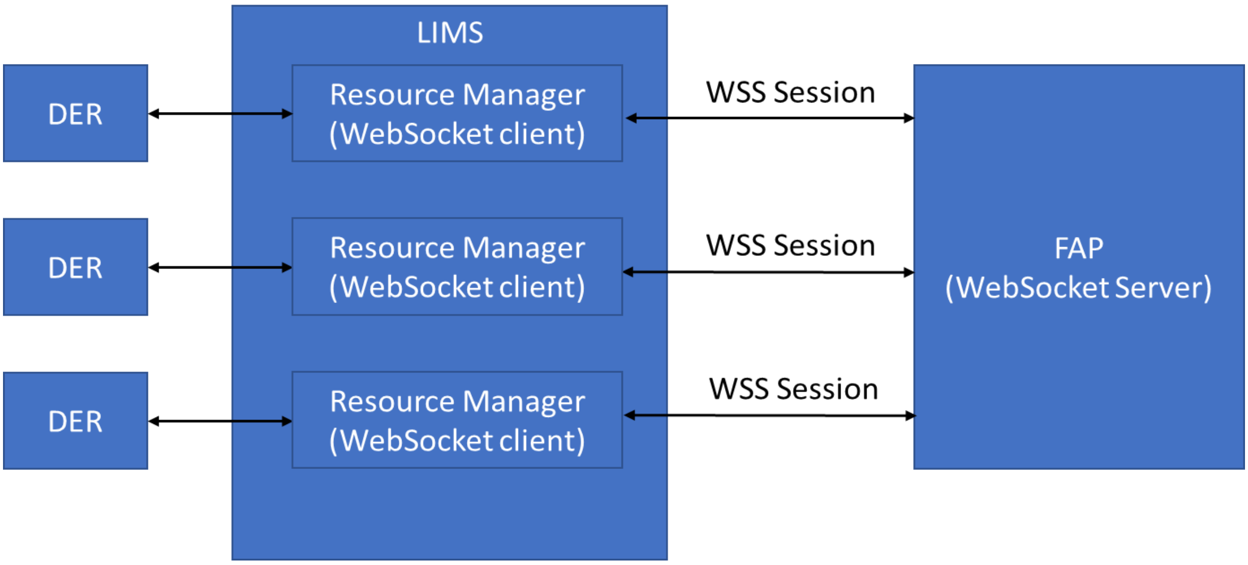

Figure 22 - EFI + communication

Initiating a connection
With WebSockets, the client initiates the connection. Each DER has its own WebSocket client instance, and exactly one WebSocket Session. This way messages of different DERs are kept separate.
In order for the client to connect with the server, the client needs to have a URL to connect to. The FAP can create a unique URL for every DER, or it can have one entry point and figures out in a later stage which DER it is. Either solution is fine.

Security
In order to secure the connection, TLS with server certificates shall be used. This concept is known as 'WebSocket Secure' (WSS), which is a parallel to HTTPS. The usage of WebSocket Secure implies that URLs will start with 'wss://'. The usage of WSS ensures that communication is encrypted between client and server, and that the identification of the server is provided through a certificate. The certificate should be signed by a Certificate Authority (CA). This can, for example, be achieved by user Let's Encrypt.
In order identify the client, basic authentication shall be used. With basic authentication, a username and password will be send by the client to the server through an HTTP header.

The username and password are not encrypted in the header, but since the connection is encrypted using WSS we consider this save enough.
The FAP should, amongst the URL, provide a unique username and password for each DER. The username must be identical to the resourceId used within the EFI+ messages.

Error handling
In general, when an error occurs, the client or the server should close the WebSocket session. When a session is closed, a Status code can be send to other side. De following table will indicate when which status code will be used. There is also a possibility to provide a reason, which is formatted as a string. This can be used as a human-readable way of giving further information of what went wrong.

Type of error	Session closed by	Status code	Further info
Internal error	Server	1011	Something went wrong on the server side
Internal error	Client	4000	Something went wrong on the client side
Resource is going offline	Client	1001	
Resource already connected	Server	4001	The old connection should be closed with this status code, the new connection should be accepted
Received illegal EFI+ message	Server	4002	For example: not valid XML, does not comply with XSD, reference to non-existing identifier…
Received illegal EFI+ message	Client	4002	See above

Reconnection strategy
Whenever the WebSocket session is closed, or when the server is unreachable, the client should attempt to reconnect. Attempting to reconnect should be done until reconnection is successful.
In order to reconnect quickly in the case of a short disruption, but to avoid flooding when the system is offline for a long time, a simple exponential back-off function shall be implemented:
Delay before next reconnection attempt in seconds = min((2 ^ *number of failed attempts*) / 2), 600)

6.3. Appendix 3: Operator user interfaces of LIMS

The user interface (UI) of the LIMS can be accessed by logging in to:
https://www.energyresourcemanager.nl/interflex
In **Erreur ! Source du renvoi introuvable.**23, the UI is illustrated:

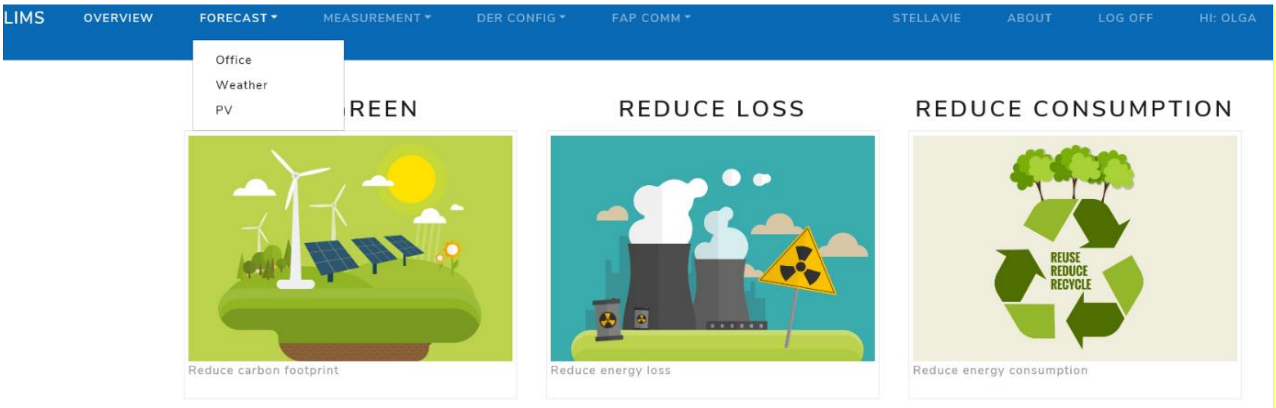

Figure 23 – Overview LIMS user interface

It has multiple functionalities:

- Overview: Provides current (real time) measurement data and instructions of the office, the SSU and the PV system. Provides the option to switch between FAP-systems , switch between maintenance, local or remote mode and the ability to reset the websocket connection.
- Forecast: Provides a graphic illustration of the forecast, as well as a comparison between the forecast and the measurement of the previous day. This is done for the office, weather and the PV-system.
- Measurement: Provides actual measurement data for the office, weather, PV and SSU systems. The data is provided in a graphical illustration as well as a data table.
- DER config: A table with parameters, parameter values, used to configure the static data in the EFI+ messages and the configuration of the communication with the FAP and DER.
- FAP communication: A logging of all EFI-messages sent to or provided by the FAP-system. The log contains the message ID, datetime, message, message type, application, log type, source and destination.
- Stellavie: This functionality provides insight in the state-of-charge of the Index solar vehicle and, with the right authorization, provides the possibility to control the charging process.

Erreur ! Source du renvoi introuvable.24 shows an example of the forecast for the energy consumption of the Eindhoven office, as displayed in the UI. Other graphs resulting from the UI will have a similar structure and lay-out.

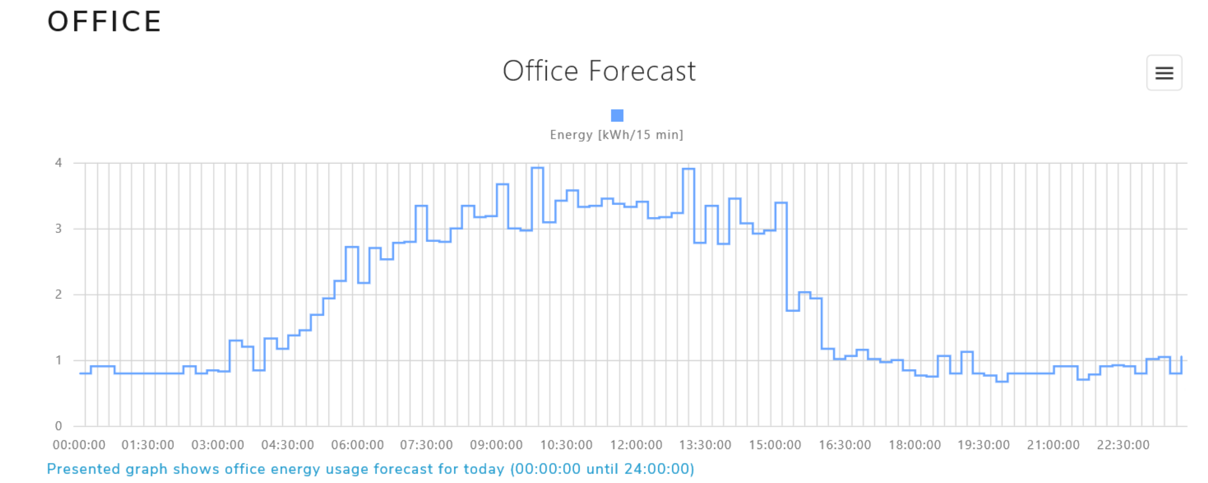

Figure 24 - Office forecast, as presented in the UI under "Forecast"

Erreur ! Source du renvoi introuvable.25 shows an example of the overview that is provided by the UI. As can be seen, the power is displayed for the office, the SSU and the PV-system.

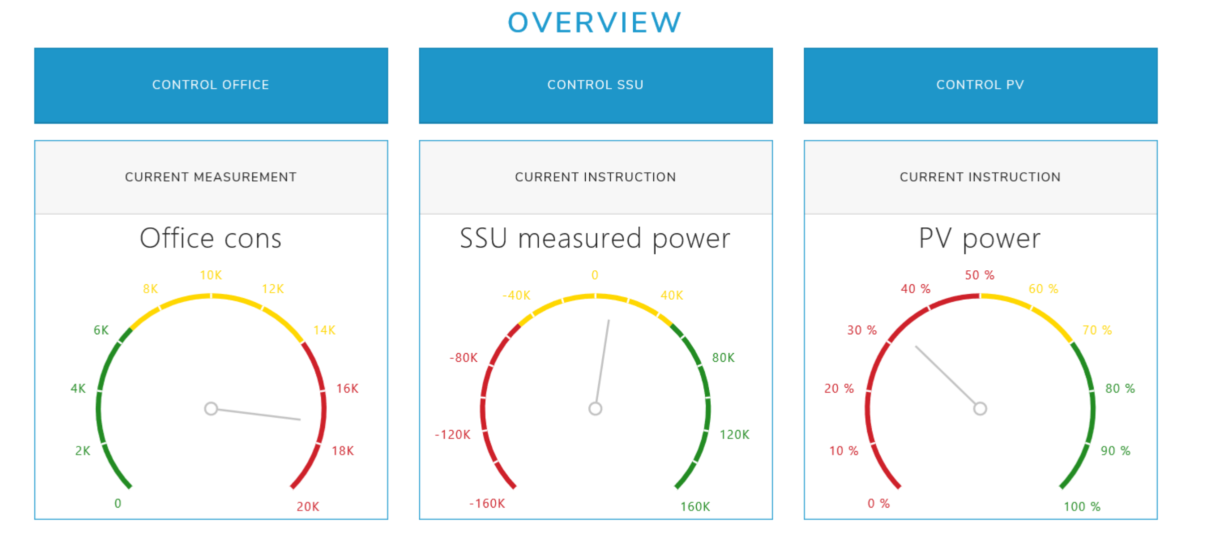

Figure 25 - Current measurements/instructions as presented in the UI under "overview"

6.4. Appendix 4: EFI background

Why EFI?

Currently, there are many Demand-Side Management (DSM) solutions available that exploit the flexibility of energy devices, such as USEF, OpenADR, Triana and TNO's own PowerMatcher and HeatMatcher. All of these solutions initially used their own way to model the flexibility and communicate with the managed energy device, resulting in many different protocols (and often mixing protocols and algorithms). Since there are many devices, this doesn't scale well: device vendors need to support many DSM solutions and DSM vendors need to support all possible devices. Furthermore, buying one (silo) solution will create a vendor lock-in, restricting consumers in freely choosing their DSM and device combinations. This limits the adoption speed of exploiting flexibility, delaying the energy transition.

EFI objectives:

1. interoperability between demand response services and smart devices
2. accelerating innovation by preventing DSM silo solutions / vendor lock-in
3. provide a solid base for future developments of DSM solutions and/or smart devices
4. simplify architectures of smart grid solutions by separation of concerns

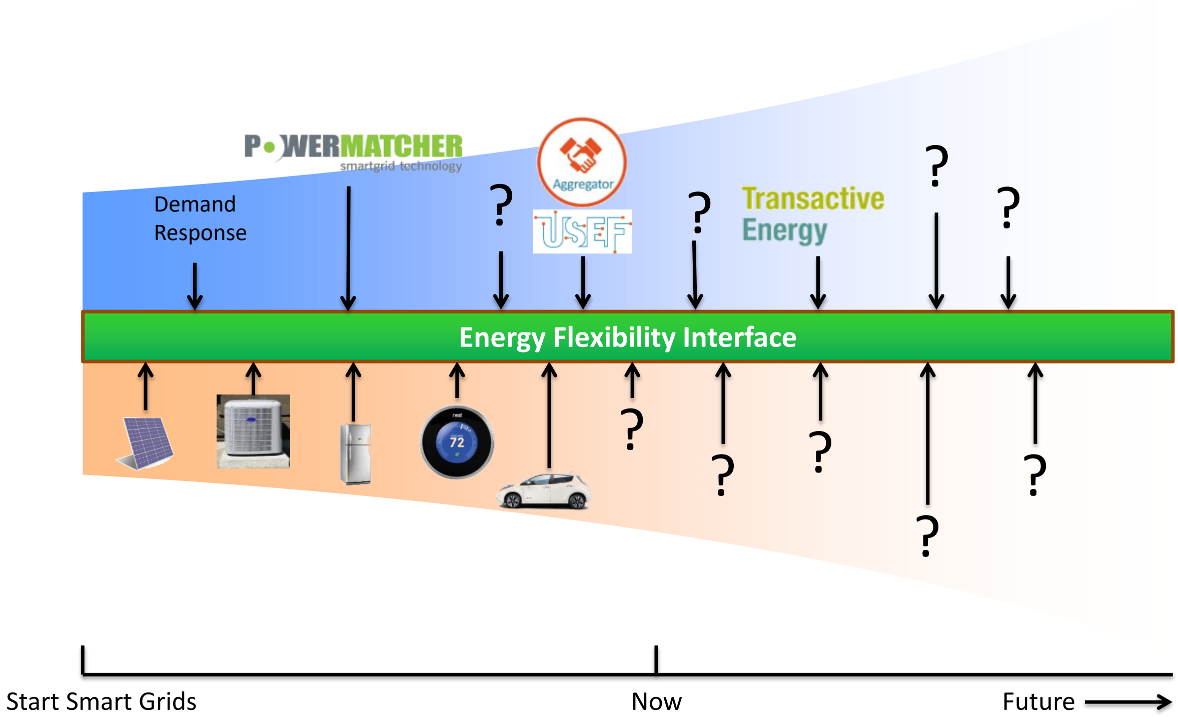

Figure 26 - Energy Flexibility Interface

We don't know what DSM solutions will appear nor what smart devices will become available. What we do know, is that we want to exploit their flexibility in a simple and uniform way.

How does EFI work?

EFI introduces interoperability in the communication between device and DSM. TNO has thoroughly analysed the information exchange between device and DSM and has defined four different device categories to model energy flexibility. They provide an abstraction of the devices regarding energy flexibility and are device and DSM independent. The four categories are as follows:

Inflexible	Cannot be controlled and has no actual flexibility, but is measurable and may provide forecasts	*Static Photo-voltaic panels, domestic loads, windmills, solar collectors*
Shiftable	Process which can be shifted in time, e.g. has a deadline	*Washing machines, dryers, dishwashers*
Storage	Flexible in production / consumption level, but is bounded by a buffer. Deadlines and required fill levels constrain the flexibility of this category.	*Freezers, CHPs, thermal buffers, stationary batteries, electrical vehicles*
Adjustable	Flexible in production / consumption level and not constrained by a buffer. They have a wide range of control possibilities without many restrictions and therefore usually offers a lot of flexibility.	*Rotating Photo-voltaic panels Generators, dimmable lighting, heat pumps, gas heaters*

EFI as a common language for energy flexibility

EFI creates a common language for energy flexibility (as depicted in **Erreur ! Source du renvoi introuvable.**7), allowing any combination of device and DSM-solution. It is designed to be future proof and only provides an abstraction of the device for modelling energy flexibility (e.g. it does not solve issues regarding home automation).

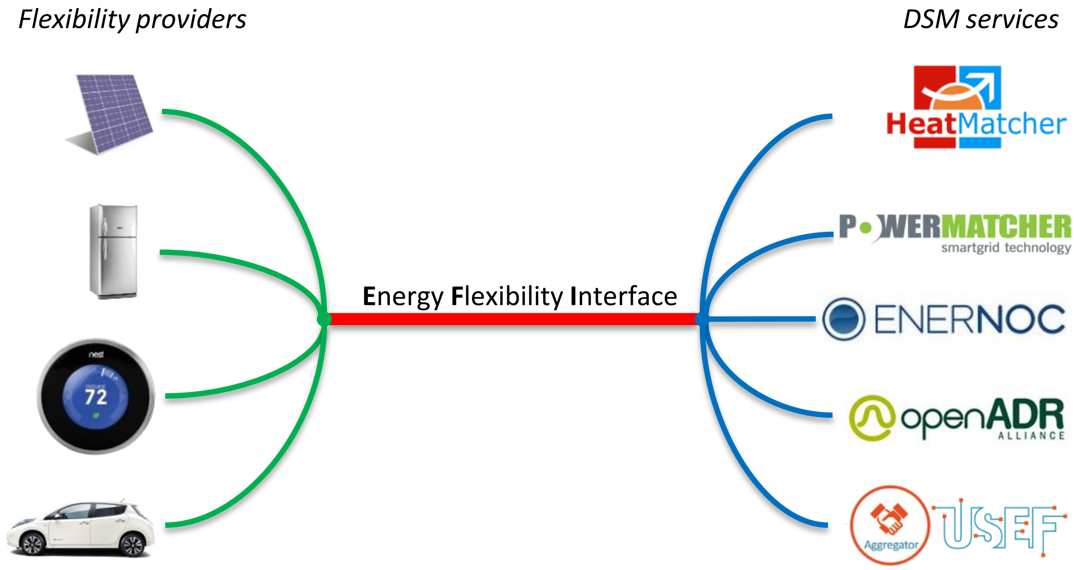

Figure 27 - EFI provides one common language for energy flexibility.

This allows all devices to communicate with all Demand-Side Management solutions without having to develop custom adapters for each combination. It creates an open playing field for all parties regarding flexibility services.

Flexibility Interface for Aggregators
Besides device energy flexibility, higher level aggregated flexibility (such as flexibility of cold stores, multiple households and industrial processes) can be modelled using EFI with the Aggregator category of EFI. This category is an extended version of the Shiftable category and supports flexibility models required for trading on day-ahead, intra-day, and real-time energy markets by aggregators.

Governance by the Flexible Power Alliance (FAN)
EFI is available as open source standard which is currently governed by the FAN foundation. Currently, EFI concepts are being standardized within the European standardization organization CENELEC, to become a European Standard for modelling energy flexibility.

More information
More information can be found at the FAN foundation website at: http://flexible-energy.eu/.
Developers wishing to experiment with the models can download them from our GitHub repository at https://github.com/flexiblepower. There you can also find a reference implementation using the EF-Pi software platform of TNO. The interface is also available as an XSD with XML examples.
A detailed description of the models can be found in the EFI whitepaper.

6.5. Appendix 5: Modbus Interface

Modbus interface to SSU

The choice was made according to the documentation standard for the selected type of battery. Modbus TCP is connected to the LIMS via a 4G VPN router installed at the SSU.

Erreur ! Source du renvoi introuvable.8 shows the signal list that is used for the Modbus protocol.

		Input Registers:	
Register Address	ID	Description	Units
30101	101	General Status	bit defines
30102	102	'Last set power value'	kiloWatts (signed 16 bits)
30103	103	Status Inverter	bit defines
30105	105	Status BMS 1	bit defines
30106	106	Status BMS 2	bit defines
30107	107	Status BMS 3	bit defines
30108	108	Status BMS 4	bit defines
30115	115	Battery State of Charge	(0-100) % (16 bits unsigned)
30116	116	Container energy meter KW (High Reg)	kiloWatts (32 bit float IEEE 754)
30117	117	Low Reg of Container energy meter KW	
30118	118	Container energy meter KWH IN	kWh (32 bit float IEEE 754)
30119	119	Low Reg of Energy meter KWH IN	
30120	120	Container energy meter KWH OUT	kWh (32 bit float IEEE 754)
30121	121	Low Reg of Energy meter KWH OUT	
30122	122	Energy meters communication status	bit defines
30123	123	Actual container temperature	Celcius (signed 16 bits)
30124	124	Actual Reefer control status	bit defines
30201	201	DD EMS date/time in mastercontroller	day
30202	202	MM	month
30203	203	YYYY	year
30204	204	HH	hour
30205	205	MM	minutes
30206	206	SS	seconds
30207	207	Software version data : part 1	2 x char
30208	208	Software version data : part 2	2 x char
30209	209	Software version data : part 3	2 x char
30210	210	Software version data : part 4	2 x char
30211	211	Custom Inputs	bit defines
30251	251	Container energy meter Volt phase 1	V (32 bit float IEEE 754)
30252	252	PART 2 of 32 bits float	
30253	253	Container energymeter Volt phase 2	V (32 bit float IEEE 754)
30254	254	PART 2 of 32 bits float	
30255	255	Container energymeter Volt phase 3	V (32 bit float IEEE 754)
30256	256	PART 2 of 32 bits float	
30257	257	Container energymeter Current phase 1	A (32 bit float IEEE 754)
30258	258	PART 2 of 32 bits float	
30259	259	Container energymeter Current phase 2	A (32 bit float IEEE 754)
30260	260	PART 2 of 32 bits float	
30261	261	Container energymeter Current phase 3	A (32 bit float IEEE 754)
30262	262	PART 2 of 32 bits float	
30263	263	Container energymeter frequency	Hz (32 bit float IEEE 754)
30264	264	PART 2 of 32 bits float	

Figure 28 – Protocol for input Registers

Modbus interface to PV panels
The Modbus protocol was already standard supported from both the existing Softs PLC and the LIMS system. Modbus TCP is connected to the LIMS via a 4G router.
In order to connect the PV, a Virtual Private Network (VPN) is established. This connection is established by an OpenVPN software connection. (https://openvpn.net/)

In figure 29 the signalling is shown and figure 30 gives a overview of the user interface to the PV panels/

Sturing motor 1	1 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 1	2 R	Input register	Bit 0= Zonnepanelen Bit 1=Reclame Bit 2=Luchtzuivering
Sturing motor 2	3 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 2	2 R	Input register	Bit 3= Zonnepanelen Bit 4=Reclame Bit 5=Luchtzuivering
Sturing motor 3	5 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 3	2 R	Input register	Bit 6= Zonnepanelen Bit 7=Reclame Bit 8=Luchtzuivering
Sturing motor 4	6 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 4	2 R	Input register	Bit 9= Zonnepanelen Bit 10=Reclame Bit 11=Luchtzuivering
Sturing motor 5	7 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 5	2 R	Input register	Bit 12= Zonnepanelen Bit 13=Reclame Bit 14=Luchtzuivering
Sturing motor 6	8 W	Holding Register	1= Zonnepanelen 2=Reclame 3=Luchtzuivering
Feedback motor 6	4 R	Input register	Bit 0= Zonnepanelen Bit 1=Reclame Bit 2=Luchtzuivering
Actuel vermogen in W	9 R	Input register	
Dag opbregst in Wh	10 R	Input register	

Figure 29 - The signal list that is used for the Modbus protocol

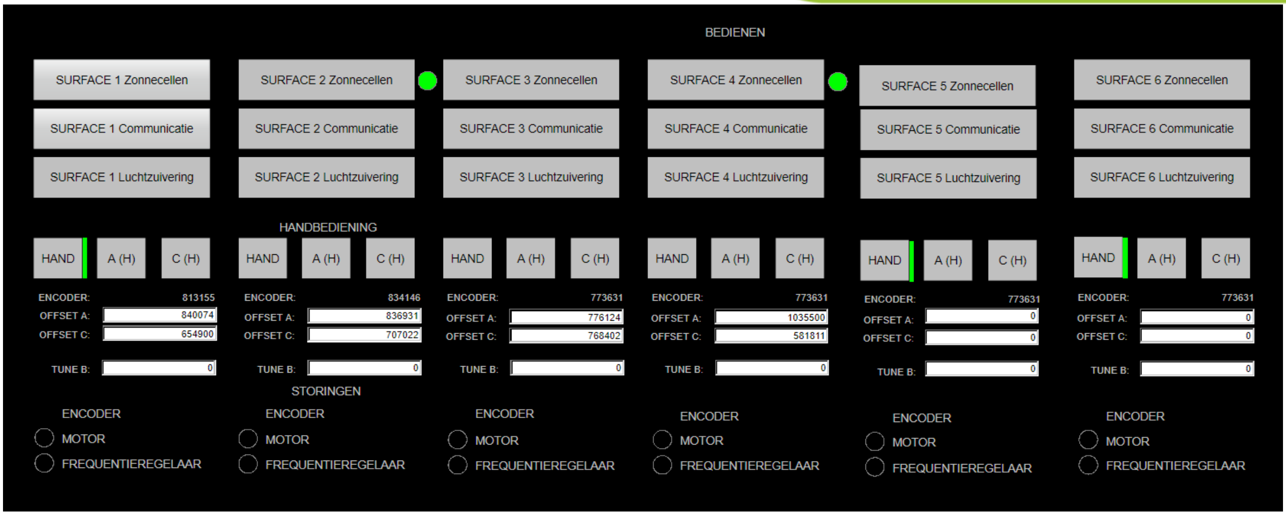

Figure 30 – Panels overview

Modbus interface to solar car

The connection from the charging station to the LIMS is enabled by a wireless internet connection through a Virtual Private Network (VPN) with a 4-G router - connection.

The LIMS system uses the Energy Resource Manager software engine trough a web socket technology in combination with the Modbus protocol and 4G technology to control the bidirectional charging station and Stella Vie vehicle.
The users can also monitor or, if needed, manually control the Stella Vie solar powered family car by using the secure LIMS mobile web portal.
The historical forecast, measurement and control data is stored in a central database to make the forecast more accurate over time and do in depth analysis.

Modbus interface to CroonWolter&Dros building

The existing energy meter has a flickering LED indication of energy usage. Via a measurement transducer this LED is read via an optical sensor and connected to Modbus TCP. The Modbus TCP is connected by cable to the central LIMS server also located in the building.