Deliverable 4.5.2
Tool for flexibility interface

August 29th 2019

EcoGrid 2.0 is a research and demonstration project funded by EUDP (Energiteknologisk Udviklings- og Demonstrationsprogram). The 9 partners in the project are:

Main Authors:

Name/Partner	Email
Troels Brødsgaard, Uptime	troels@uptime.dk

TABLE OF CONTENTS

1 Introduction

In order for the flexibility market described in EcoGrid 2.0 to become a success, it is important to ensure interoperability between the distributed energy resources (DERs) which the aggregators control in order to deliver flexibility to the market. Interoperability serves two purposes:

1. Minimize the amount of investment aggregators must make in order to include many different DERs in their portfolio. This lowers the barrier-to-entry for aggregators and maximizes the potential size of the aggregators DER portfolio.
2. Avoid aggregator lock-in for end-customers. Interoperability allows end-customers to choose their aggregator freely, no matter their type of DER.

This report describes the work done in terms of interoperability requirements description and specification during the EcoGrid 2.0 project.

Initial work focused on defining the requirements for interoperability in a commercial setup. The output from this was a set of business cases with high-level descriptions of different interaction scenarios between aggregators and DERs. These are described in section 2 (Protocol requirements).

After the requirements had been described and distilled, they were used to evaluate existing standards which could be suitable for implementing interoperability. A number of different standards were identified with a background in either grid, factory or home automation. This is described in section 3 (Standard evaluations).

While the surveyed standards were founded on many good principles, none of the evaluated standards were found to be directly applicable to the EcoGrid 2.0 demonstration setup. As a consequence, a new standard, the Ecogrid Interoperability Protocol (EIP), was informally described. The EIP is described in section 4 (Protocol description).

For the purpose of EcoGrid 2.0 it was only required to implement a subset of the EIP. This was due to the fact that the project only had one kind of DER represented: home energy management systems (HEMSs), some of which had very little capability in terms of remote control. The implementation of the EIP for EcoGrid 2.0 is described in section 5 (Implementation in EcoGrid 2.0). This final section also introduces the Flexibility Interoperability Platform, which was built to translate messages between the aggregators and the HEMSs. This allowed aggregators to control the HEMSs using the EIP.

2 Protocol requirements

During the first year of the EcoGrid 2.0 project, significant effort was spent on describing the foundational requirements of the ICT tools to be developed. Among these were the tools for interoperability. The requirements were expressed as business cases, which described the essence of the different tools and their interactions from a high-level and non-technical point of view.

The requirements for interoperability were specified in two business cases, “BC01 - Contracting Flexibility Representation” and “BC04 - Interoperability and Control of Flexibility During Operation”. Together, they identified the scenarios listed in Table 1~~Table 1~~.

Scenario name	Scenario description
BC01	
Acquisition of flexibility representation.	The process of the DER owner acquiring an active aggregator for being represented in the market as a flexibility provider.
Change of flexibility representation	The process of the DER owner changing active aggregator.
Exit from flexibility representation	The process of the DER owner exiting the market as flexibility provider.
BC04	
Real-time control	The aggregator sends control messages to the DER to be effectuated immediately.
Aggregator-initiated control coordination	The aggregator sends a coordination message to the DER, The DER evaluates the coordination message and returns a response which indicates some form of commitment.
DER-initiated control coordination	The DER sends a coordination request to the aggregator. The aggregator then begins the coordination flow in scenario 2.
Triggered execution of coordinated control	The DER evaluates and applies a control action according to prior coordination, and optionally sends an action report to the aggregator.
State information exchange triggered by aggregator request	The aggregator sends a request for DER state information. The DER responds with information on its current state.
State information exchange triggered by DER event	An event internal to the DER triggers the DER to send information on its current state to the aggregator.
DER-initiated general information exchange	The DER requests a piece of general information from the aggregator. The aggregator responds with the information.
Aggregator-initiated general information exchange	The aggregator sends a piece of general information to the DER.

Table 1 - List of scenarios identified by business cases

The message flow required for each scenario was identified and described as sequence diagrams and step-by-step scenario descriptions. These can be found in the business cases in appendices 6.1 (BC01 Contracting Flexibility Representation~~BC01 Contracting Flexibility Representation~~) and 6.2 (BC04 Interoperability and Control of Flexibility during operation~~BC04 Interoperability and Control of Flexibility during operation~~).

The focus during the formulation of these scenarios was the technical breadth of devices which can act as DERs in some regard. Some devices may simply expose the functionality of a switch, letting the aggregator control whether it is switched on or off, while other devices may embed sophisticated control logic able to optimize their operation according to pricing signals, weather forecasts or forecasted energy production or consumption, etc.

Through describing these scenarios, a few requirements for the technical/communication setup were identified:

- Default "dummy" aggregators which a DER can connect to when not under contract with an actual aggregator. Dummy aggregators facilitate the first joining of a DER into aggregator representation, as well as leaving representation in case the contract is cancelled or the aggregator goes out of market.
- The aggregator must be able to learn the capabilities of a DER when representation is requested. This means the DER must itself be involved in the request for representation.
- All aggregators must host a directory listing of all aggregators, allowing a DER to discover available aggregators.
- Communication between the aggregators and DERs must be asynchronous and bidirectional, allowing both aggregators and DERs to send unsolicited messages to each other, in near real-time fashion.

All of these requirements influenced the work on interoperability in EcoGrid 2.0, though the scope for implementation was reduced. In EcoGrid 2.0, all DERs were assumed to be represented by one of the two aggregators in the project. As such, implementation did not need to consider dummy aggregators. Also, there was no contractual negotiation as part of representation. When participants wanted to change which aggregator represented them, this was always implicitly accepted, and the aggregator was simply notified of the new DER.

3 Standard evaluations

After describing the requirements, a brief survey of already established standards was conducted, and each standard was evaluated in relation to the described requirements. There are already several established standards under the smart grid umbrella, all of which have a starting point in some related domain, be that either grid operations, factory automation or home automation. Due to their distinct starting points, they all have unique advantages and disadvantages. The following is a listing of the different standards which were surveyed, as well as a quick summary of the result of the evaluation.

3.1.1 IEC 60870

IEC 60870 defines systems used for telecontrol in regard to electrical engineering and power system automation. While still in widespread use, especially in Europe, it is considered a legacy standard to be superseded by the newer IEC 61850. It suffers from several connectivity issues, such as net security issues and the inability to traverse NAT. As such, it was not considered further.

3.1.2 IEEE 1815 (DNP3)

Work on DNP3 was performed in parallel with the work on IEC 60870 and the two protocols have several things in common. DNP3 is the dominant protocol used for telecontrol of substations in North America today, but it had a wider scope of implementation and is also used in many other domains, including oil and gas, water/wastewater, environmental and security industries.

In 2003, DNP3 was adopted by the UK water industry for use in the WITS (Worldwide Industrial Telemetry Standards) protocol. This shows the ability of the DNP3 standard to evolve outside its initial scope. However, the WITS group has since identified issues in regard to scaling WITS-DNP3, leading to the development of WITS-IoT, which was announced in 2016. WITS-IoT is set to address two issues with WITS-DNP3: product development cost and connectivity. WITS-DNP3 is not suitable for quick product development, nor for low cost products due to the high cost of implementing the standard. Further, as DNP3 uses the same connectivity technology as IEC 60870, it suffers from the same issues.

The first version of the WITS-IoT protocol was announced in October 2018, which was too late for use in EcoGrid 2.0. Further, the standard is only available to WITS members. The main features of WITS-IoT is the definition of a JSON schema for expressing the DNP3 application layer and the use of MQTT to provide connectivity.

3.1.3 IEC 61850

IEC 61850 is an international standard defining communication protocols for intelligent electronic devices at electrical substations. While IEC 61850 can be seen as a successor to both DNP3 and IEC 60870, it has a much wider scope of standardization. In addition to communication methods, it also defines an object oriented information model for logical representation of substation elements and functionality as well as a substation

configuration language (SCL) which assists in the engineering process of substation devices.

While the primary focus of the IEC 61850 standard is electrical substations, it has been extended to several other domains such as wind power plants, hydroelectric power plants and distributed energy resources through additional technical specifications. The SCL supports the rigorous engineering process involved in design, procurement and production of high-value assets such as power plants and substations, which lowers the associated costs.

The standard introduces novel communication concepts for low latency communication over Ethernet such as GOOSE and SMV, enabling messages to be sent and received in 4 milliseconds on the local process bus, which is fast enough for tripping protection devices. Communication over the Internet is done using the MMS (Manufacturing Message Specification) application layer protocol.

The extensibility of the information model makes it suitable for expressing the domain treated in EcoGrid 2.0, and the SCL could be useful for communicating the capabilities of DERs. However, IEC 61850 is an even bigger standard than WITS-DNP3, and as such it may not be suitable for mass-produced consumer-facing products.

One interesting note is that IEC 61850 has recently been extended with 61850-8-2:2018. This new part of the standard specifies a method of exchanging data through any kinds of network, including public networks, using XMPP and end-to-end secured communications. This enables its use in more smart grid use cases, and could be applicable to a project like EcoGrid 2.0.

3.1.4 IEEE 2030.5 (Smart Energy Profile 2.0 or SEP2)

The SEP 2.0 protocol is designed as a protocol that can connect smart energy devices in the home to the smart grid. It allows smart energy devices on the same local network to discover each other, and to utilize the functions of the other devices for their operation. It has functions defined for metering, pricing and load control, and its data objects are based on the information model of IEC 61850. It is intended for use in Home Area Networks (HANs) and requires an Energy Service Interface for interaction between HAN and aggregator.

3.1.5 OpenADR

OpenADR is a standard developed by the OpenADR Alliance for demand response. It has seen some adoption in the US. The standard uses the exchange of price signals as a means to regulate demand. It does not cover other means of regulating demand, and as such was not considered further.

3.1.6 OPC UA

OPC Unified Architecture is a relatively new, open standard for industrial automation and process control, authored by the OPC Foundation. It represents a unification of previous OPC standards and uses modern and widely accessible technologies in its

implementation specifications. In contrast to previous OPC standards, OPC UA is platform independent, making it applicable in a much wider range of scenarios.

OPC UA is also not constrained to industrial automation. Its comprehensive information modelling framework allows for the modelling of complex, nested structures, much like those in IEC 61850. Companion specifications have been developed to map OPC UA to other domains, such as oil and gas, packaging and robotics. In 2018, a companion specification mapping OPC UA to IEC 61850 was released.

Of further interest, the OPC Foundation released a specification extension called "OPC UA PubSub" in 2018. This extension adds soft real-time bi-directional communication patterns using publisher-subscriber technologies to OPC UA, which are very applicable in IoT scenarios like EcoGrid 2.0.

OPC UA has been translated into Woopsa, the "Web Object Oriented Protocol for Software and Automation". Woopsa was quite influential in the formulation of the Ecogrid 2.0 Interoperability Protocol, as described in sections 4 (Protocol description) and 5 (Implementation in EcoGrid 2.0).

4 Protocol description

The Ecogrid 2.0 Interoperability Protocol (EIP) was developed as part of EcoGrid 2.0 in order to provide an interoperability protocol suitable for the needs and constraints of the project. The protocol leans on principles from the surveyed protocols and is structured to match the functional requirements identified in section 2 (Protocol requirements). The protocol description is not very detailed or specific, as it covers a much larger set of functionality than was required in EcoGrid 2.0. Details on those parts which were implemented in EcoGrid 2.0 can be found in section 5 (Implementation in EcoGrid 2.0).

The protocol description is split in two parts: information model and services. The information model describes how device capabilities are modelled while the services describe message exchanges used to realise the scenarios described in section 2 (Protocol requirements).

4.1 Information model

The EIP information model is based on concepts from Woopsa, which in turn is based on the OPC UA information model. These concepts inform the message structure used in EIP, as well as the modelling of DER capabilities.

In EIP, the functionality of DERs is described using a tree of logical nodes. The tree starts from a node named **Root** which serves as a well-known entry point. The tree of nodes below the root node represent the capabilities of the DER. The tree structure allows aggregators to discover the capabilities of an unknown DER through tree traversal, and the tree of a DER can be composed of the capabilities of multiple connected devices. For instance, a local EIP gateway could combine the capabilities of an e-meter, EV charger and heat pump controller.

Every node in the tree can have both properties and methods associated with it, and these are used to learn or affect the status of the capability. The properties of a node reflect the current status of the capability. Properties may be either writeable or read-only. Read-only properties cannot be changed by aggregators. Aggregators can invoke capabilities of DERs by either by changing a writeable property or by calling a method.

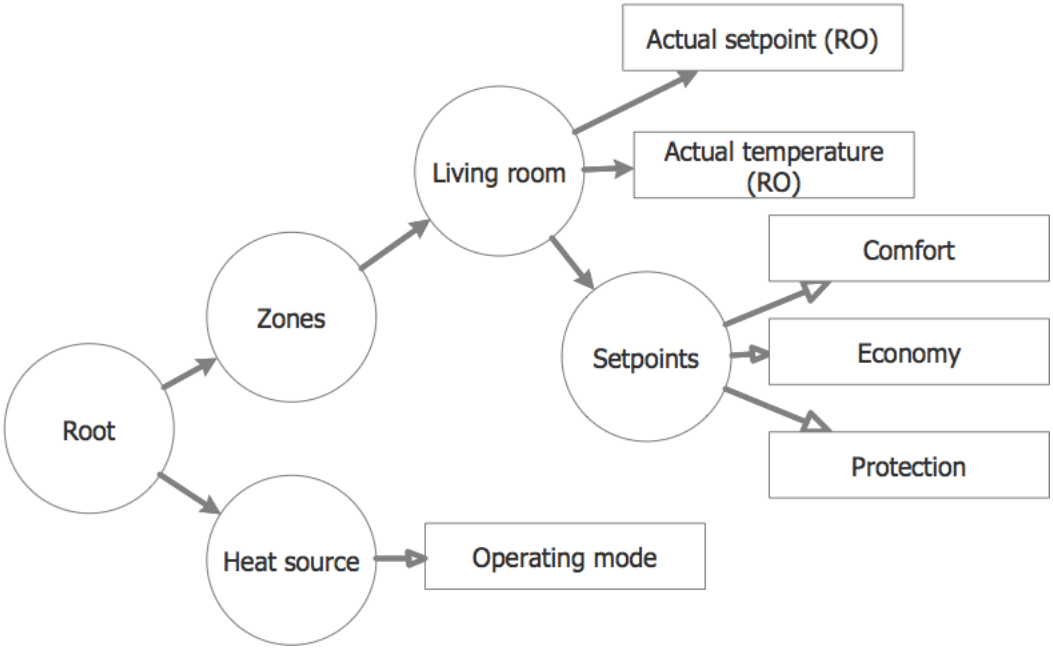

Figure 1 – Partial tree model of an example HEMS

Figure 1~~Figure 1~~ shows part of a tree which models a DER based on household heating. The root node has two child nodes - **Zones** and **Heat source**. The **Heat source** node has a single property **Operating mode**, which exposes the capability to control the operating mode of the household heat source.

The **Zones** node itself has a child node **Living room.** The **Living room** has two properties **Actual setpoint** and **Actual temperature** which are both read-only (**RO**). It also has a child node **Setpoints**, which has properties for **Comfort**, **Economy** and **Protection** setpoints.

Every element in the tree can be referenced through its associated path. The path to an element is a concatenation of the names of all elements between it and the root element, using "/" as a separator between element names. The name of the root node is not included in the path. As an example, the path to the Heat Source node is "/Heat Source" and the path to the Protection property of the Setpoints property of the Bathroom is "/Zones/Bathroom/Setpoints/Protection".

The value type of a property can be either a simple value (float, integer, string, etc) or a composite value (object or array).

4.2 Services

The scenarios described in section 2 (Protocol requirements) outline several message flows, some of which have commonalities across the scenarios. In the EIP, these message flows have been formalised as high-level services. A service describes message flows for information exchanges, by defining the order and types of messages to be exchanged by aggregator and DER. There are five services: Information, Representation, Control, Coordination and State. The actual content of the messages is described more fully in section 5 (Implementation in EcoGrid 2.0) for the implemented services.

4.2.1 Information

The information service is used by the aggregator to share general (non-control) information with the DER. The aggregator achieves this by sending an information **message**. The DER can send an information **request** to trigger information exchange. Among other things, it can be used by the DER to request the directory of aggregators, and for the aggregator to inform the DER that it has been transferred to a different aggregator.

4.2.2 Representation

The representation service is used by a DER to request representation from an aggregator. The DER sends a representation **request** to the aggregator, which responds with a **receipt**. No further information exchange is required for representation – if the end-customer and the new aggregator enter into a representation agreement, the DER will receive a notice from its current aggregator via. the information service.

4.2.3 Control service

The control service allows the aggregator to perform real-time control of a DER. The aggregator sends a control **message** to the DER, which is acknowledged by the DER with a **receipt**. The DER performs an action equivalent to the control message, and sends a **report** to the aggregator.

4.2.4 Coordination service

The coordination service allows aggregators and DERs to coordinate control in advance. Coordination can be initiated by both aggregator and DER, and contains both a coordination and execution phase.
The coordination phase is started by the aggregator sending a coordination **message** to the DER, which is acknowledged by the DER with a **receipt**. The DER then evaluates the content of the coordination message, and sends a **result** message to the aggregator.
The execution phase is triggered by the DER when the pre-conditions for the coordinated action are satisfied. The DER then performs the given control action and sends a **report** to the aggregator with the result of the control action.
If the DER wants to initiate coordination, it can do so by sending a coordination **request** to the aggregator. The aggregator must then enter the coordination phase.

4.2.5 State service

The state service allows aggregators to learn the state of a DER. The aggregator may trigger a state exchange by sending a state **request** to the DER, which will reply with a **message**. The DER can also trigger state exchange by sending an unsolicited **message**.

5 Implementation in EcoGrid 2.0

While the EIP covers a broad spectrum of possible interactions between aggregator and DER, the interactions required for EcoGrid 2.0 could be covered by a subset of the EIP. For this reason, implementation was only performed for a subset of the protocol. This section describes how the functionality of EcoGrid 2.0 equipment was mapped to the EIP information model, what technologies were chosen for message exchange and how the EIP services were expressed using these technologies.

5.1 Mapping the EIP information model to EcoGrid 2.0

EcoGrid 2.0 based its demonstrations on two different home energy management systems: Greenwave Reality and Siemens Synco Living. The two types of HEMS implement several divergent features, but both allow for some measure of remote control over household heating through an application programming interface. In EcoGrid 2.0 it was decided to focus on direct real-time control of household heating. Thus the HEMS features related to this aspect were mapped to the EIP.

Houses which had Greenwave Reality systems installed were either heated by a heat pump or heat panels. Remote control of heating in these systems was done by controlling the state of a relay. When the relay was switched on, the heat source was able to regulate itself to provide heat according to parameters configured in the device. When the relay was switched off, the heat source was not allowed to provide heat. There was only one relay per household, which means it controlled heating in the entire household.

All houses with a Siemens Synco Living HEMS were heated by heat panels. Here, remote control was done by changing the active temperature setpoints. The temperature setpoints were controlled per room.

There were two main goals with implementing the EIP for these two systems:

1. Show the ability of the protocol to harmonize differences in remote control capability, in order to expose a common control interface from both types of devices.
2. Show the ability of the protocol to express multiple levels of capabilities in a single HEMS.

The control functionality of the Siemens HEMS can be seen as a superset of that provided by the Greenwave HEMS. With the Siemens HEMS, simultaneously changing the setpoint in all rooms to the lowest possible temperature has the same effect as switching the relay off in the Greenwave HEMS. Conversely, changing the Siemens HEMS setpoints in all rooms back to their normal settings is the same as switching the relay on in the Greenwave HEMS.

Thus, the common control interface exposed for Greenwave and Siemens HEMS was modelled as shown in Figure 2~~Figure 2~~.

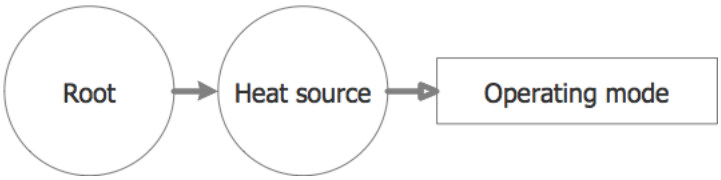

Figure 2 – The information model for the common control interface

The common control interface is exposed through the **Heat source** node, which has an **Operating mode** property. The value of the **Operating mode** can be either **Auto** or **Off**. In **Auto**, the HEMS provides heating according to its internal parameters. In **Off**, the HEMS is not able to provide heat.

In addition to the common control interface, the ability in the Siemens HEMS to control setpoints per room was also mapped to the EIP. This mapping was a direct translation of part of the Siemens HEMS functionality to the EIP. Appendix 6.3 (FIP API) contains documentation for this part of the implementation (see chapter 5, Siemens Native Control Interface).

5.2 Choice of technology

When choosing technology for the implementation, the most important criteria was scalability. Both in terms of using widespread, well understood technology that would be readily available to the implementors in EcoGrid 2.0, but also in terms of scaling to a real world commercial setup.

The MQTT v3.1.1 protocol was chosen to provide asynchronous, bi-directional communication between DERs and aggregators. MQTT is a publish-subscribe-based messaging protocol where clients communicate through a central broker. The protocol is popular in the “Internet of Things” ecosystem, as it is lightweight and optimized for small sensors and actuators and communicating over high-latency and unreliable networks. It was also already used in the implementation of the EcoGrid market platform.

While MQTT specifies how messages are transported, it does not dictate the payload of those messages. As with the information model, the message payloads were based on Woopsa. Woopsa uses JSON to encode its message payload. The JSON format is very widely used and supports encoding of all data types used in EIP. JSON is also used in the implementation of the EcoGrid market platform.

5.3 Implementing EIP services using MQTT and JSON

The only EIP services implemented in EcoGrid 2.0 were the state and control services. With these services, the aggregators were able to traverse the tree of a device to discover its capabilities, read the current state of its capabilities and change the state. This subsection will describe the common parts of the implementation and the service-specific parts for the state and control services.

5.3.1 General

In MQTT, all messages must be sent to a specific topic. The message topic was used to identify the service and message type for a given message, as well as the device that the message pertained to. The general form of MQTT topics were: “device_id/service_name/message_type”.

The payload of all messages exchanged as part of the protocol followed a common format. All messages contained a header and body. The header was the same for all message types, while the body varied according to the message type. The message body could be interpreted according to the fields specified in the header.

As mentioned, the payload was encoded as JSON. An example payload could look like this:

```
{
  "Header": {
    "Verb": "meta",
    "CorrelationId": "db0fba1f-2934-4848-98e4-a1d55bf1fad8",
    "SentAt": "2017-08-24T18:49:26",
    "Ttl": 300
  },
  "Body": {
    "Path": "/"
  }
}
```

The header keys are described in Table 2~~Table 2~~.

Key	Type	Description
Verb	JSON string	The Woopsa verb (see Table 3~~Table 3~~).
CorrelationID	JSON string	UUID used to correlate messages belonging to the same message flow.
SentAt	JSON string	UTC timestamp in ISO8601 format, indicating when the message was sent.
Ttl	JSON number	Message time to live, expressed as a number of seconds (when the message is older than “SentAt + Ttl” it will be discarded)

Table 2 - Header fields

Note that the fields CorrelationId, SentAt and Ttl are left out for brevity in the examples below. They must always be present in the header.

Verb	Description
meta	Used to describe nodes in the tree in order to traverse and discover device functionality.
read	Used to read the value of a property.
write	Used to write the value of a property.

Table 3 - Woopsa verbs and their usage in EIP

5.3.2 State

As described in section 4.2.5 (State service), the state service uses two different message types: **request** and **message**. This service is used by the aggregator to traverse the tree of a device, as well as to read the value of properties.

Sending a message with the verb "meta" with the path to a node returns the list of items (child nodes) and properties connected to the node. Example message exchange are listed in Table 4~~Table 4~~.

Message type	Topic	Payload

Request (Aggregator to FIP)	device_id/state/request	{
		"Header": {
		"Verb": "meta"
		},
		"Body": {
		"Path": "/"
		}
		}

Response (FIP to aggregator)	device_id/state/message	{
		"Header": {
		"Verb": "meta",
		},
		"Body": {
		"Name": "/",
		"Items": [
		"Zones",
		"Heat source"
		],
		"Properties": []
		}
		}

Table 4 – Describing the root node

To read the value of a property, one must send a message with the verb "read" and the path to the property. Example message exchange to read the current value of the property "/Heat source/Operating mode" is listed in Table 5~~Table 5~~.

Message type	Topic	Payload

Request (Aggregator to FIP)	device_id/state/request	{
		"Header": {
		"Verb": "read"
		},
		"Body": {
		"Path": "/Heat source/Operating mode"
		}
		}

Response (FIP to aggregator)	device_id/state/message	{
		"Header": {
		"Verb": "read"
		},
		"Body": {
		"Type": "String",
		"Value": "Auto",
		"ReadOnly": false
		}
		}

Table 5 – Reading the current value of "/Heat source/Operating mode"

5.3.3 Control

As described in section 4.2.3 (Control service), the control service uses three different message types: **message**, **receipt** and **report**. This service was used by the aggregator to change the value of properties.

To change the value of a property, a message must be sent with the verb "write" and the the path to a property. Example message exchange to change the value of the "/Heat source/Operating mode" property is listed in Table 6~~Table 6~~.

Message type	Topic	Payload

Request (Aggregator to FIP)	device_id/control/message	{
		"Header": {
		"Verb": "write"
		},
		"Body": {
		"Path": "/Heat source/Operating mode",
		"Value": "Off"
		}
		}

Receipt (FIP to aggregator)	device_id/control/receipt	{
		"Header": {
		"Verb": "write"
		},
		"Body": null
		}

Report (FIP to aggregator)	device_id/control/report	{
		"Header": {
		"Verb": "write"
		},
		"Body": {
		"Type": "String",
		"Value": "Off",
		"TimeStamp": "2019-08-24T18:49:27"
		}
		}

Table 6 - Changing the value of "/Heat source/Operating mode"

5.4 Flexibility Interoperability Platform

In order to translate from the EIP to the native protocol of the HEMSs in EcoGrid 2.0 it was decided to build the Flexibility Interoperability Platform (FIP). As such, the FIP sat as a mediator between the aggregators and the HEMSs.

The FIP exposed an MQTT broker for the aggregators to connect to. The aggregators would send messages to this MQTT broker according to the above specification. When a message for a device is received, it must be translated and dispatched to the HEMS using the API provided by the HEMS vendor.

The FIP would use the EcoGrid data warehouse to determine which HEMSs were represented by which aggregator. It would also subscribe to the relevant MQTT topics in order to react to aggregator messages, and publish responses from the devices back to the aggregator.

The FIP must be able to reliably dispatch to the entire collection of HEMSs within a short period of time, as simultaneous flexibility activations of both aggregators entire portfolio are to be expected. The FIP must be able to handle this load, and dispatch requests to the HEMSs in an optimal way.

In addition, the FIP should attempt to accommodate for temporary communication failures with the HEMSs. In many cases this can be achieved with a simple retry strategy, possibly using increasing back-off intervals.

Finally, the FIP should strive for safety. In order not to compromise participant comfort the FIP must implement recovery strategies in case of communication failure between the FIP and HEMSs.

6 Appendices

6.1 BC01 Contracting Flexibility Representation
See “BC01_Contracting_Flexibility_Representation.docx”

6.2 BC04 Interoperability and Control of Flexibility during operation
See “BC04_Interoperability_and_Control_of_Flexibility_during_operation.docx”

6.3 FIP API
See “FIP_API.pdf.”

Read more at www.ecogrid.dk