Connecting Smart Grid Protocol Standards
A Mapping Model Between Commonly-used Demand-response Protocols OpenADR and MIRABEL

Sevket Gökay, Markus C. Beutel, Houran Ketabdar and Karl-Heinz Krempels
RWTH Aachen University, Information Systems, Ahornstr. 55, 52074 Aachen, Germany

Keywords: Mapping Model, OpenADR, MIRABEL, Interoperability.

Abstract: Heterogeneous smart grid systems operate with different and incompatible protocols. MIRABEL and OpenADR are prominent examples, providing intelligent demand respond functionalities. In principle of operation as in complexity, both protocols differ significantly, which results in a lack of inter-connectivity among themselves. Connecting these commonly used standards makes it possible to benefit from different protocol advantages and prevents from reconstructing whole smart grid systems for consolidation. Furthermore, it holds potentials for interoperability of individually produced smart grid components. This work contributes a conceptual mapping model between OpenADR and MIRABEL on the basis of a detailed protocol analysis, as well as an initial implementation.

1 INTRODUCTION

Smart grids leverage communication technology to act on information about the behaviors of power suppliers and consumers (Farhangi, 2010). Related functionalities are provided via different sorts of protocols. As a result of different protocol standards, there is a lack of inter-connectivity among heterogeneous smart grid systems. Developing a mapping tool, which connects the prominent protocols OpenADR and MIRABEL, prevents from changing the whole system structure in case of a grid consolidation. Above that, realization of interoperability between smart grid components offers significant economic potentials for both manufacturers and users.

In a first step, we study the modeling of OpenADR and MIRABEL to find similarities and differences in roles, processes and information exchange concepts. In a nutshell, we will find out that there are three comparison levels: Operation/service, operation message and message field. Operation/service level similarities and differences will be semantic, and the remaining two will be mostly syntactic. Afterwards, we prototype a mapping tool, which connects the mentioned standards and describe early evaluation results.

The work is structured as follows. Section 2 explains related mapping efforts. Moreover, Section 3 gives an overview of demand response protocols, before Section 4 describes the contributed approach and Section 5 presents the current status of the implementation and the evaluation. Finally, Section 6 reflects the contributions and describes upcoming future work.

2 RELATED WORK

This section gives a general impression of related work done in the field of interest.

(Ghatikar et al., 2014) concentrated on the mapping between constitutive OpenADR versions. Basically, this work helps to get insights about the technical compatibility of each version. Furthermore, (McParland, 2011) focused on the development of an implementation of OpenADR for smart grids. (Fischer et al., 2013) describes efficiency aspects of MIRABEL. (Koch and Piette, 2007) enable a wider range of facilities to leverage the benefits of demand response. (GWAC, 2008) proposes a general framework to organize concepts and terminology for interoperability issues. This framework basically focuses on a high, organizational level.

(Rumph et al., 2013) classify existing technologies to enable the interoperability of energy management systems. Moreover, they outline a conceptual model, which focuses on the semantic level of interoperability.

As a preliminary conclusion, there is no solution

Gökay S., C. Beutel M., Ketabdar H. and Krempels K..
Connecting Smart Grid Protocol Standards - A Mapping Model Between Commonly-used Demand-response Protocols OpenADR and MIRABEL.
DOI: 10.5220/0005486503820387
In *Proceedings of the 4th International Conference on Smart Cities and Green ICT Systems* (SMARTGREENS-2015), pages 382-387
ISBN: 978-989-758-105-2

that enables interoperability among the examined protocols on both the semantic and the syntactic level.

3 DEMAND RESPONSE PROTOCOLS

Demand response (DR) is a technology developed to cope with the increasing energy demands without increasing the energy generation (Mashima et al., 2014). DR plays an important role to manage and optimize volatile loads of renewable energy sources. It enables to maintain a balance between power generation and load distribution with demand, without any manual intervention. This is usually realized by a communication protocol between the energy provider and consumer in which the endpoints negotiate the electricity demand, price and respective time periods.

This section gives a brief overview of two such DR protocols, OpenADR and MIRABEL. For further explanations, the reader is advised to look at respective resources, e.g., specifications.

3.1 OpenADR

Open Automated Demand Response (OpenADR) facilitates sending and receiving DR signals from a utility or independent system operator to electricity customers (Zuber et al., 2013). It is led by North American research labs and companies. Principally, it is a two-way signaling system between a server (*Virtual Top Node - VTN*) and client (*Virtual End Node - VEN*). VTN publishes DR signals to clients and VEN controls the electric energy demand in response to DR signals. The VEN may be either a producer or consumer of energy. It is possible for a VEN to take on the role of VTN to distribute DR signals (Figure 1). These nodes are called aggregators.

The latest version OpenADR 2.0b specifies four application-ready services, on which we focus in this work:

1. Registration (EiRegisterParty): VENs have to be registered by VTNs before any other interaction.
2. Event (EiEvent): This is the main service of the protocol to communicate DR information models and event functions for price signals. Events are generated and maintained by VTNs and sent to VENs for them to accept.
3. Reporting or Feedback (EiReport): Defines periodic or one-time information on the state of a resource.

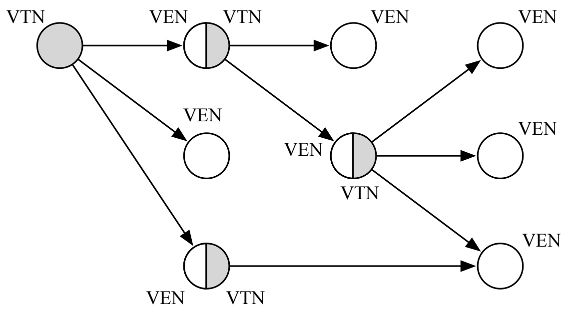

Figure 1: Possible relationship between VENs and VTNs.

4. Opt or Override (EiOpt): Communicates opt-in and opt-out schedules based on short-term changes in availability of a resource.

It also identifies following services, planned for future releases: Enrollment, market contexts, quote or dynamic prices, availability.

On a technical level, OpenADR specifies data models (defined in XML schemata) for messages to be transmitted via either simple HTTP or XMPP transport.

PUSH/PULL Operations. OpenADR can be used in a PULL mode, where the VEN asks for updates from the VTN, or in a PUSH mode, where the operation is initiated by the VTN to a VEN.

3.2 MIRABEL

MIRABEL stands for Micro-Request-Based Aggregation, Forecasting and Scheduling of Energy Demand, Supply and Distribution. It is intended to balance the available supply of renewable energy sources and current demand (Verhoosel et al., 2011). It is supported by the European Commission's Seventh Framework Program (FP7)[1].

MIRABEL revolves around the *flexibility* concept of electricity demand and supply. This concept defines that producers or consumers *(prosumers)* can offer possible changes to time periods and energy amounts (i.e. flexibilities) of both load generation and consumption (Figure 2). In this regard, *flexOffer* messages are exchanged indicating these flexibilities. Another actor is the Balance Responsible Party (BRP). The BRP is responsible of aggregating the *flexOffers*, scheduling them based on market situation, availability of energy, etc. It negotiates the price, the use and timing of *flexOffers* with the prosumers (Verhoosel et al., 2011). The BRP can accept an offer and send a *flexOfferAcceptance*. This only signifies that the BRP is going to make use of it at some point in time (within

[1]http://ec.europa.eu/research/fp7

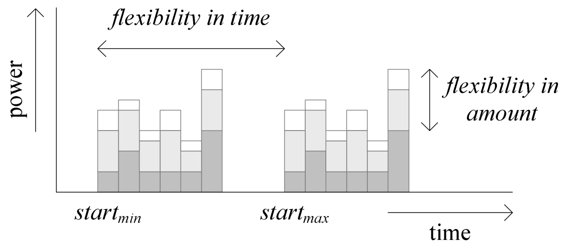

Figure 2: Flexibility in MIRABEL with dimensions time and energy amount (Verhoosel et al., 2011).

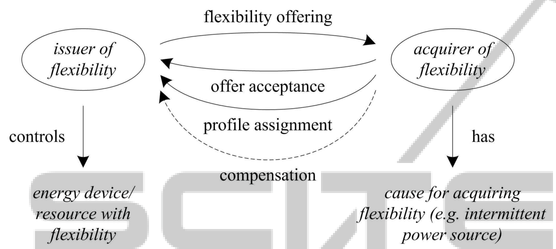

Figure 3: Roles in MIRABEL (Verhoosel et al., 2011).

the defined time frame in *flexOffer*). The actual decision of energy usage profile (time, energy, costs) is communicated with a *flexOfferAssignment* message afterwards. In Figure 3, prosumer is depicted as the issuer of flexibility and BRP as the acquirer of flexibility. The specification mentions many more roles of energy domain that can interact via MIRABEL, but the focus is on prosumer and BRP.

In summary, MIRABEL only defines one service with *flexOffer* being the request and *flexOfferAcceptance* or *flexOfferAssignment* the responses. On a technical level, MIRABEL specifies data models (defined in XML schemata) for messages but not the underlying transportation protocol.

3.3 Protocol Comparison

At this point, we don't want to discuss protocol characteristics in close detail, but rather describe similarities and differences which are most relevant for upcoming mapping efforts. An in-depth analysis leads to following conclusions:

1. Since MIRABEL does not specify a transport protocol, we can accept simple HTTP communication as a common ground.
2. Both standards define their data models in XML schemata and use XML messages.
3. Both use similar roles, which allow a general mapping without extensive modifications in this area (Table 1).
4. OpenADR defines four services whereas MIRABEL only one. The OpenADR services EiRegisterParty and EiReport have no logical equivalents in MIRABEL. See Section 4.2 for workarounds.
5. OpenADR defines its data types in much more detail than MIRABEL.
6. Whereas in MIRABEL, prosumers send their flexibilities and BRP can accept that or assign a new energy usage/schedule, in OpenADR VTNs distribute possible events to VENs and VENs can decide on accept/opt-in/opt-out. So, there is a conceptual difference that the deciding role is reversed.

Table 1: Comparison of MIRABEL and OpenADR roles.

MIRABEL role	*Corresponding OpenADR role*
Consumer	VEN
Producer	VEN
BRP	VTN

This situation shapes the relation between both protocols: OpenADR has a comprehensive feature set and MIRABEL is a simpler protocol in comparison. Many operations and messages of OpenADR do not exist in MIRABEL or the differences are so big that a workaround cannot be applied. Therefore, it is only possible to map from MIRABEL to OpenADR but not the other way around. In other words, our aim is to make MIRABEL OpenADR-*compatible*.

4 MAPPING

After having established the direction of mapping (MIRABEL to OpenADR), we can now look at the fine-grained comparison of message types. The intention is to bridge the gap between MIRABEL's one service and OpenADR's EiEvent and EiOpt services since they convey DR information. Figure 4 illustrates the general idea of message flow and the function of our tool as a MIRABEL BRP and OpenADR VEN to mediate between MIRABEL prosumers and OpenADR VTNs.

Communication with the Prosumer

The mapping tool accepts *flexOffer* messages, processes and maps them to OpenADR messages, communicates with a VTN and based on the outcome of the communication sends *flexOfferAcceptance* and *flexOfferAssignment* to the prosumer. This is basically the working principle of the solution.

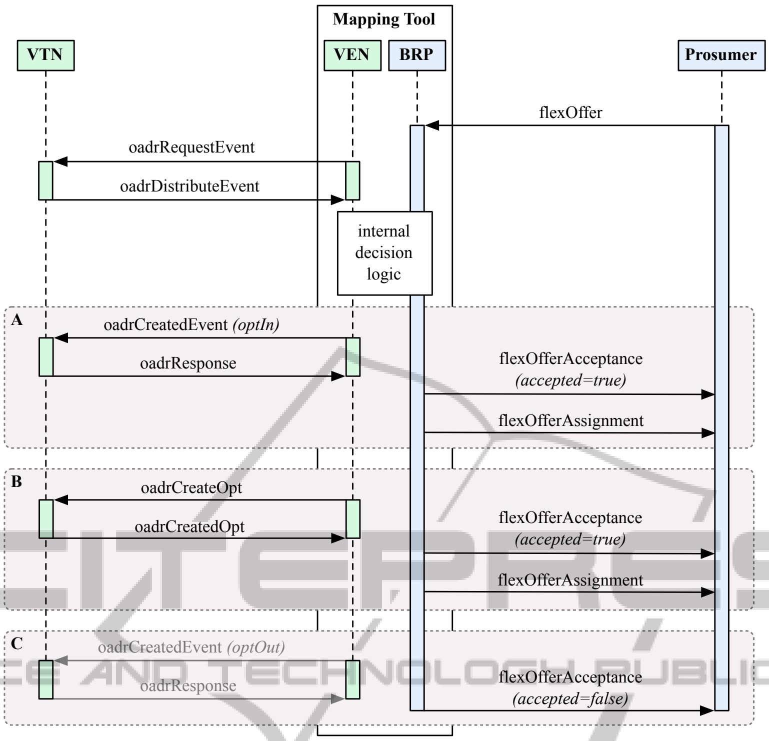

Figure 4: Overview of message flows.

Communication with the VTN

As mentioned, we have a work flow mismatch: *flexOffer* proposes a schedule and/or energy usage, but a VEN can only ask for events (schedules) with either *oadrPoll* (periodically) or *oadrRequestEvent* (one-time) messages in PULL mode. Both message types serve the goal to acquire applicable events (*oadrEvent*) from VTN which is realized by *oadrDistributeEvent* message. Afterwards, a VEN can decide to opt-in or opt-out events and transmit the decision.

The Decision Logic

The way the mapping tool works is upon receiving *oadrEvents* from VTN, it compares the schedules, energy amounts and pricing from these events with those from the received *flexOffer*.

A. If the flexibility offer is within acceptable bounds, i.e. one of the events completely overlaps the offer, it sends *oadrCreatedEvent* with `optIn` state to VTN and *flexOfferAcceptance* with `accepted` field set to `true` to prosumer (Figure 4A).
B. If there is only a partial overlap, it overrides one OpenADR event, depending on the factor which requires the least extension, by sending *oadrCreateOpt* and accepts the *flexOffer* (Figure 4B).
C. If there is no match, it sends either no message or (if the event requires a response) *oadrCreatedEvent* with `optOut` state to VTN, and *flexOfferAcceptance* with `accepted` field set to `false` to prosumer (Figure 4C).

In any case, if the reply from VTN indicates that request cannot be accepted, *flexOfferAcceptance* with `accepted` field set to `false` is sent to prosumer.

4.1 Mapping of Fields

The two elements, which contain DR information, in aforementioned messages are FlexEnergy and EiEvent. This section provides an overview of the mapping of the fields within the MIRABEL data model to those in the OpenADR data model.

Trivial fields (such as IDs of messages, involved parties, etc. and some meta information) can be mapped and handled easily. Other fields, including time periods, pricing and amount of energy have to be derived and their mapping is worth mentioning.

First of all, in accordance with its flexibility concept MIRABEL allows the use of both single values and range values (min-max, upper-lower, after-before) rather than just single values as in OpenADR. Therefore, in case of a range, the OpenADR value has to be always checked whether it is contained within

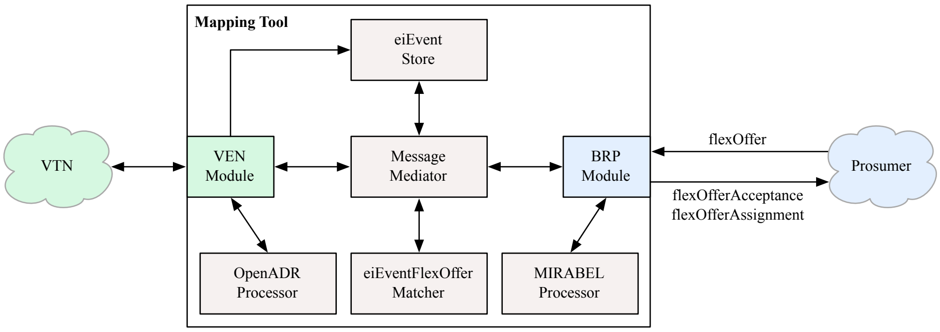

Figure 5: Components overview of the mapping tool.

the bounds:

$$min\_value_{MIR} \leq value_{OADR} \leq max\_value_{MIR}$$

More granular mapping principles are summarized in Table 2. It should be mentioned, that predefined EiEvent signals from (Zuber et al., 2013) are advisory and deployments can define their own custom signals which facilitates the interoperability between both protocols.

4.2 Workarounds

Since MIRABEL does not support querying the actual energy usage, we cannot provide a workaround for EiReport. On the other hand, we propose that the mapping tool tracks the prosumers it communicates with and stores them in a database. After receiving a *flexOffer* from a new, unknown prosumer, it automatically registers the prosumer using the EiRegisterParty service by the VTN. If the VTN decides to cancel registration anytime, the mapping tool can mark this prosumer as canceled and subsequent *flexOffers* will not be accepted. In that way, the tool deals with EiRegisterParty internally without outside interference.

5 IMPLEMENTATION

We started developing the mapping tool as a Java server application running on Jetty[2] and leveraging RestEasy[3] for Web services and clients. The XML messages are exchanged using HTTP POST. We used OpenADR XML schemata downloaded from the OpenADR alliance website and MIRABEL XML schemata extracted from the documentation.

An overview of the architecture can be seen in Figure 5. The *modules* consist of client and service parts and are responsible for communicating with the remote endpoints via the Internet. The *processors* serialize/deserialize requests and responses. In order to reduce the remote calls to VTN, *eiEventStore* caches the distributed eiEvents for local lookup first. *eiEventFlexOfferMatcher* includes the decision logic and *MessageMediator* coordinates the whole operation.

For the evaluation purposes, we installed the OpenADR VTN developed by EPRI[4] and set up the mapping tool to communicate with it. Unfortunately, there is no obtainable implementation of MIRABEL. We therefore had to fall back on manually generating the *flexOffers* and analyzing the responses from the tool. In this regard, the fact that incoming and outgoing messages are validated against the respective XML schemata during runtime, helped immensely as a guidance. First evaluation runs showed expedient data exchange. Therefore, we assume a working mapping logic.

6 CONCLUSION

In this work, we studied two DR protocols, OpenADR and MIRABEL, which are used in smart grid systems and investigated their interoperability. We compared the two protocols on a semantic as well as syntactic level. Even though they aim to achieve similar goals, their different working principles and varying complexities make it hard to find an easy solution. While OpenADR specifies DR events which are defined ahead of time and communicated for the participation, MIRABEL targets near real-time negotiation of energy usage, time period and price. Moreover, the definition of multiple functionalities and complex data types of OpenADR and rather simplistic nature of MIRABEL make it impossible to come up with a complete, one-to-one mapping. Our mapping

[2]http://eclipse.org/jetty/

[3]http://resteasy.jboss.org/

[4]Electric Power Research Institute. http://sourceforge.net/projects/openadr2vtn/

Table 2: Matching of FlexEnergy and EiEvent fields.

FlexEnergy field	*EiEvent field*	*Description*
type	eiEventSignal.signalName AND eiEventSignal.signalType	Depending on the signal category it can be decided whether this signal relates to production or consumption of energy.
sourceType	-	No match
totalEnergyConstraint	SUM of eiEventSignal.currentValue	Aggregation over values of events with the signal name `CHARGE_STATE`.
totalPriceConstraint	SUM of eiEventSignal.currentValue	Aggregation over values of events with the signal name `ENERGY_PRICE`.
energyConstraintProfile	eiActivePeriod OR eiEventSignal.intervals	For the fields related to time schedule, the corresponding information can be read from eiActivePeriod or from one event.
energyConstraintProfile. EnergyConstraint	eiEventSignal.currentValue	Value of an event with the signal name `CHARGE_STATE`.
energyConstraintProfile. PowerConstraint	eiEventSignal.currentValue	Derived from events with the signal name `LOAD_DISPATCH`.
tariffConstraintProfile	eiEventSignal.currentValue	There is no explicitly declared tariff concept in OpenADR, but tariffs can be computed from values of signals with signal type `priceMultiplier`.

logic effort shows that interoperability is possible and serves as a starting point.

6.1 Limitations and Future Work

Due to the dissimilar characteristics of both protocols, the mapping is limited in operational scope. Future work primarily consists of solving the incompatibilities and broadening the extent of functionalities (e.g., reporting). In this prototypical phase, we concentrated on the functional evaluation of the mapping logic rather than other aspects such as performance and speed. In addition, we acknowledge that our evaluation lacks a sound approach and the moving parts for integration testing. These are part of future work.

ACKNOWLEDGEMENTS

This work was funded by the Excellence Initiative of the German State and Federal Government (Project Urban Future Outline).

REFERENCES

Farhangi, H. (2010). The path of the smart grid. *Power and Energy Magazine*, pages 18–28.

Fischer, U., Kaulakiene, D., Khalefa, M. E., Lehner, W., Pederson, T. B., Siksnys, L., and Thomsen, C. (2013). *Real-time Business Intelligence in the MIRABEL Smart Grid System.* Springer.

Ghatikar, G., Riess, D., and Piette, M. A. (2014). Analysis of Open Automated Demand Response Deployments in California and Guidelines to Transition to Industry Standards.

GWAC (2008). GridWise Interoperability ContextSetting Framework.

Koch, E. and Piette, M. A. (2007). Architecture Concepts and Technical Issues for an open, Interoperable Automated Demand Response Infrastructure. In *Grid Interop Forum*.

Mashima, D., Herberg, U., and Chen, W. (2014). Enhancing Demand Response Signal Verification in Automated Demand Response Systems. In *Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5.

McParland, C. (2011). OpenADR Open Source Toolkit: Developing Open Source Software for the Smart Grid. In *IEEE Power & Energy Society General Meeting*, Detroit, MI.

Rumph, F., Huitema, G., and Verhoosel, J. (2013). Towards interoperability of energy management systems using flexibility through conceptual modeling. In *IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, Ghent.

Verhoosel, J., Rothengatter, D., Rumph, F., and Konsman, M. (2011). Initial draft of the MIRABEL Standard.

Zuber, J., Herberg, U., and Bienert, R. (2013). OpenADR 2.0 Profile Specification B Profile.