Development and Application of a New Ontology in the Context of Hybrid AC/DC Grids

Alessandro Rossi
Energy Green Transition
Engineering Ingegneria Informatica
Palermo, Italy
email: alessandro.rossi@eng.it

Marzia Mammina
Energy Green Transition
Engineering Ingegneria Informatica
Palermo, Italy
email: marzia.mammina@eng.it

Jawad Kazmi
Center for Energy
AIT Austrian Institute of Technology
Vienna, Austria
email: jawad.kazmi@ait.ac.at

Zhiyu Pan
Institute for Automation of CPS
RWTH Aachen University, Aachen, Germany
email: zhiyu.pan@eonerc.rwth-aachen.de

Charles Emehel
Institute for Automation of CPS
RWTH Aachen University, Aachen, Germany
email: charles.emehel@eonerc.rwth-aachen.de

Bharath Varsh Rao
Center for Energy
AIT Austrian Institute of Technology, Vienna, Austria
email: bharath-varsh.rao@ait.ac.at

Antonello Monti
Institute for Automation of CPS
RWTH Aachen University, Aachen, Germany
email: amonti@eonerc.rwth-aachen.de

***Abstract*—The HYPERRIDE project aims to enable a unique revolution in the electrical grid infrastructure creating the conditions to really unlock a wide application of Direct Current (DC) technology in the distribution grid. By combining DC and Alternating Current (AC) technologies, HYPERRIDE will demonstrate potential solutions that are seen in AC-DC hybrid grids for Low Voltage (LV) and Medium Voltage (MV) infrastructures, as most power electronics applications use internal DC power supplies. Furthermore, HYPERRIDE provides a technology-independent specification of a FIWARE-based interoperable and secure Information and Communications Technology (ICT) platform. In this paper, after giving a quick rundown on main energy domain ontologies that share knowledge conceptualization to allow an easier systems interaction and give the system components reasoning capabilities and autonomy, the Hybrid Ac Dc Grid Ontology (HADGO) developed inside the project will be described and a real case application will be presented. The Switch Gear use case was successfully modeled and evaluated for inconsistencies using the HermiT Reasoner. Asserted and inferred facts were achieved and more use case scenarios can be updated on the HADGO ontology either through hard coding on the Protégé GUI or using the ontology learning method. A total of 301 asserted and inferred axioms were achieved using the HermiT Reasoner on the Protégé ontology development tool. The HADGO ontology is applied in the HYPERRIDE sensing and monitoring infrastructure layer. It shows the usability of HADGO ontology in a real use case scenario of a grid information system.**

Keywords-Hybrid AC/DC, Smart Grid, Ontology, ICT, Interoperability, Reasoner.

I. Introduction

The role of Distributed Energy Resources (DER) is increasing significantly in electrical power systems due to many environmental, economic, and political drivers [1]. This transition has also put the electrical distribution grid in a central role. The challenges arising from this transition are largely being addressed under Smart Grid (SG) [2] initiatives. Although there is no standard definition, in general, a SG refers to a method of incorporating intelligence into the operation of a distribution grid to increase flexibility and performance. For electrical power systems, AC distribution grids are a well-known infrastructure that has been in use for a long time. This infrastructure can be assisted by DC technologies as a possible backbone to increase, for example, Renewable Energy Sources (RES) hosting capacity; however, they must be designed on a solid basis to allow for rapid roll-out and integration. It is critical to provide and test suitable methodologies and resources to lower entry barriers for early adoption processes to maximize the implementation capability of new DC technologies. The HYPERRIDE project aims to support this transition toward the transformation in the electrical grid infrastructure by laying the groundwork for the widespread adoption of DC technology. The future distribution grid both at the Low Voltage Direct Current (LVDC) component to Medium Voltage Direct Current (MVDC) backbone is planned to be demonstrated at three pilot sites (Germany, Italy, and Switzerland) implementing relevant use cases. These pilots will provide valuable insights and help identify the gaps in knowledge and possible solutions for the various focus areas.

Interoperability among the components and sub-systems of the developed AC/DC hybrid power system solution is a key goal of the project, as having an interoperable solution has numerous benefits for all stakeholders. In general, interoperability [3] implies that information conveyed from a *sending* system to a *receiving* system can be used meaningfully by the latter, necessitating at least some interpretation and contextualization of the data. Interoperability, however, is a challenging quality attribute to achieve because, in addition to some other technical and governance challenges, it necessitates a thorough understanding of the problem, the solution, and its interrelation. Data is at the center of interoperability, necessitating its consideration in any effort to achieve a higher level of interoperability.

An ontology is a formal description of knowledge as a set of concepts within a domain and their interrelationships [4]. It provides an abstract model that can describe, in a formal language based on mathematical logic, relevant aspects (concepts, relationships, properties, facts, rules) of a phenomenon or domain of interest that is intended to be represented for some purpose. Apart from being useful for many other aspects, an ontology provides a sound basis for developing an interoperable data model that can help in the integration of SG applications. One such ontology HADGO is developed in the context of HYPERRIDE as the basis for the interoperable data models for enabling interoperability and integration of solutions in hybrid AC/DC smart grid applications.

The rest of the paper is organized as follows. Section II provides a concise review of some of the relevant ontologies and data models, Section III introduces the HADGO ontology, then in Section IV an explanation of the application of the developed ontology with some real-world use cases, is provided. Section V concludes this paper by highlighting the contribution and their effects, and also providing future research directions.

II. BACKGROUND AND STATE OF THE ART

The integration of software applications may entail substantial semantic difficulties when translating information from one application to another. Different terminologies may be used to describe the same domain and, when the same terminology is used, applications often associate different semantics with the terms. This hinders the exchange of information between applications. Ontologies may solve this issue by providing a way of explicitly specifying the semantics for each terminology unambiguously. The ontologies provide, indeed, a shared knowledge conceptualization that allows an easier system interaction and gives the system components reasoning capabilities and autonomy [5]. An ontology is a formal description of knowledge as a set of concepts within a domain and the relationships between them. It is an abstract model that describes, by using a formal language based on mathematical logic, relevant aspects (rules, properties, relationships, etc.) of the domain of interest to be represented for some purpose. Since terms and relations are shared by the entire community of the domain of interest, there is no ambiguity: an ontology describes specific knowledge unambiguously. Relationships between concepts enable automated reasoning on data, easy to implement in semantic graph databases that use ontologies as their semantic schema [6].

In this subsection, an overview of some open ontologies focused on various aspects of energy or power systems will be presented. Smart Appliances REFerence (SAREF) [7] is an ontology created to enable interoperability between smart devices. SAREF is based on the concept of a "device", which is a tangible object that we can easily find in households, public buildings, or offices and which can perform one or more functions. The SAREF ontology offers a list of basic functions that can be combined into a more complex function. Each function has some associated commands. A device can be found in some corresponding states that are also listed as building blocks. A device that wants its functions to be discoverable, registerable, and remotely controllable by other devices in the network offers a service. The service specifies the device that is offering it and its functions. A device is also characterized by an energy/power profile that can be used to optimize the energy efficiency in a home or office that is part of the building. SAREF is expressed in Web Ontology Language Description Logic (OWL-DL) and contains 124 classes, 56 object properties, and 28 datatype properties [8].

SAREF for Energy (SAREF4ENER) is Web OWL-DL ontology that is one of the many (SAREF4INMA [9]; a SAREF extension for the industry and manufacturing domain, building devices and topology [10], etc.) extensions of SAREF with new classes and properties, focusing on demand response scenarios, where customers can offer flexibility to the smart grid to manage their smart devices using a Customer Energy Manager. SAREF4ENER has been created in collaboration with Energy@Home and EEBus, which are major Italy- and Germany-based industry associations, to enable the interconnection of their different data models [11].

SmArt eneRGy dOmain oNtology (SARGON) [12] is an extension of the SAREF ontology to cross-cut domain-specific information that represents the smart energy domain. SARGON ontology is powered by smart energy standards and Internet of Things (IoT) initiatives, and real use cases. It involves classes, properties, and instances explicitly created to cover the building and electrical grid automation domain. This study exhibits the development of SARGON and demonstrates it through a web application to cross-cut domain-specific information that represents the smart energy domain and is powered by smart energy standards and IoT initiatives, as well as real use cases. SARGON involves classes, properties, and instances explicitly created to cover the building and electrical grid automation domain. The SARGON ontology network consists of several interconnected domain ontologies related to the smart grid and building automation:

- Person, Company, Building, and Address ontologies contain data for describing the nature of a person, company, building, and address, besides spaces and geometrical data such as area, place, floors, etc.;
- Device inherits all classes of SAREF ontology and extends it according to energy equipment which includes industrial equipment, energy generators, and system resources, such as Phasor Measurement Units, Proportional-Integral-Derivative controllers, converters, etc.;
- Services provides ontologies for services in the smart grid and building automation like controlling, monitoring, and protection;

 ISBN: 978-1-68558-089-6

- Common Information Model (CIM) and International Electrotechnical Commission (IEC) 61850 present terms and relations in the power grids. It identifies the list of classes and variant instances that can be used for monitoring and controlling smart grids according to the standards.

The ontologies of the SARGON network have been harmonized to enable data portability for different applications in smart energy systems including building automation and power grid monitoring and controlling. Those ontologies are intended to be used together with FIWARE Next Generation Service Interface Linked Data (NGSI-LD), a standard defined by European Telecommunications Standards Institute (ETSI) Industry Specification Group for cross-cutting Context Information Management [13].

OntoMG [14] is an ontology-based information model for microgrids that aims to solve interoperability issues (syntactic and semantic) encountered between microgrid components. It is compliant with the CIM and the IEC 61850 standards. OntoMG integrates six packages, each related to a specific aspect involved in the achievement of the microgrid objectives:

- The identification aspect (Id) consists of associating a unique identity for each stakeholder enabling an easier component recognition and implicit information extraction
- The operation aspect (Op) aims at optimizing the network operations
- The mobility aspect (Mob) captures component displacements during their lifetime
- The economical aspect (Eco) aims at minimizing total costs while considering the components' participation in the Energy Market
- The ecological aspect (Ecolo) is related to the component participation in/on the environment
- Multi-roles aspect is related to the component roles during his operation in the system.

Semantic ontologies have been proposed by several research projects and initiatives to represent data related to the energy domain used by different Energy Management Systems deployed in different smart grid scenarios, such as smart homes, urban environments (e.g., buildings, districts, cities, etc.), organizations, microgrids or Virtual Power Plants and Demand Response management. Ontology for Energy Management Applications (OEMA) [15] is an attempt to unify existing heterogeneous ontologies that represent different energy-related data. The OEMA ontology network is made up of eight interconnected domains and each ontology represents one or various energy domains: Infrastructure Ontology, Energy and Equipment, Geographical, External factors, Person and Organisation Ontology, Energy Savings, Smart Grid Stakeholders, Person and organization, Units of Measurement. These ontologies are connected by a core Ontology Network.

III. HYPERRIDE AC/DC GRID ONTOLOGY

The HADGO is developed to help in defining, modeling, and analyzing a hybrid AC/DC power grid that can then be used in different use cases in the context of the H2020 HYPERRIDE project and beyond. Some of these use cases that were considered during the formulation of the ontology include but are not limited to power-flow calculation, cascading effects calculation, critical components identification, etc. However, the demonstration of such applications is beyond the scope of this document.

An overview of the HADGO is presented in Figure 1. The figure is very detailed and shows not only the entity classes and their relationships but also highlights the object properties that are used for such relationships, as well as the entity class that is the domain and range of these properties. The ontology contains around 50 asserted entity classes. From these entity classes, Figure 2 highlights (half) the classes at the top two layers.

Additionally, the ontology contains around 186 axioms with 116 logical axioms with several object and data properties. Adopting the naming conversion usually used in ontology authoring, a prefix of `hadgo` is used with all the members of the HADGO ontology making the name following the format `hadgo:<member>`.

Developing data models in a diverse and uncoordinated manner typically results in textitdata stovepipes issues, which can lead to a total/partial failure in attaining interoperability and impede the ontology's reusability potential while also making integration difficult. When developed appropriately, ontologies can help in achieving consistency in the usage of terms and meaning leading towards achieving a common understanding and further avoiding frequent adoptions [16].

Ontological realism [17], is being advocated as one of the best practices [16] [18] for ontology development. It refers to developing an ontology to be more like a reality model than a data model to maximize its utility and stability. In this development method, the resulting ontologies serve as representations of the entities to which the data pertains rather than the data itself.

Furthermore, in the formulation of ontology, the principle of *single inheritance* is used. According to this concept, in ontology, each entity class must be a subclass of precisely one other entity class, and anything belonging to a parent term also belongs to all child terms at lower levels. This means that each asserted taxonomy has a single root node and that each ontology has one or more asserted taxonomies as suitable components.

Keeping this background in mind, the HADGO is being developed using ontological realism as well as the single inheritance rules. The knowledge is derived from some experimental and benchmark hybrid AC/DC grid models in conjunction with expert judgments, and opinions from the involved experts.

A. Entity Classes

An introduction to top-level entity classes, as highlighted in Figure 2, is provided in this section. The classes include `hadgo:PowerGrid`, `hadgo:Component`, `hadgo:ComponentType`, `hadgo:FunctionType`,

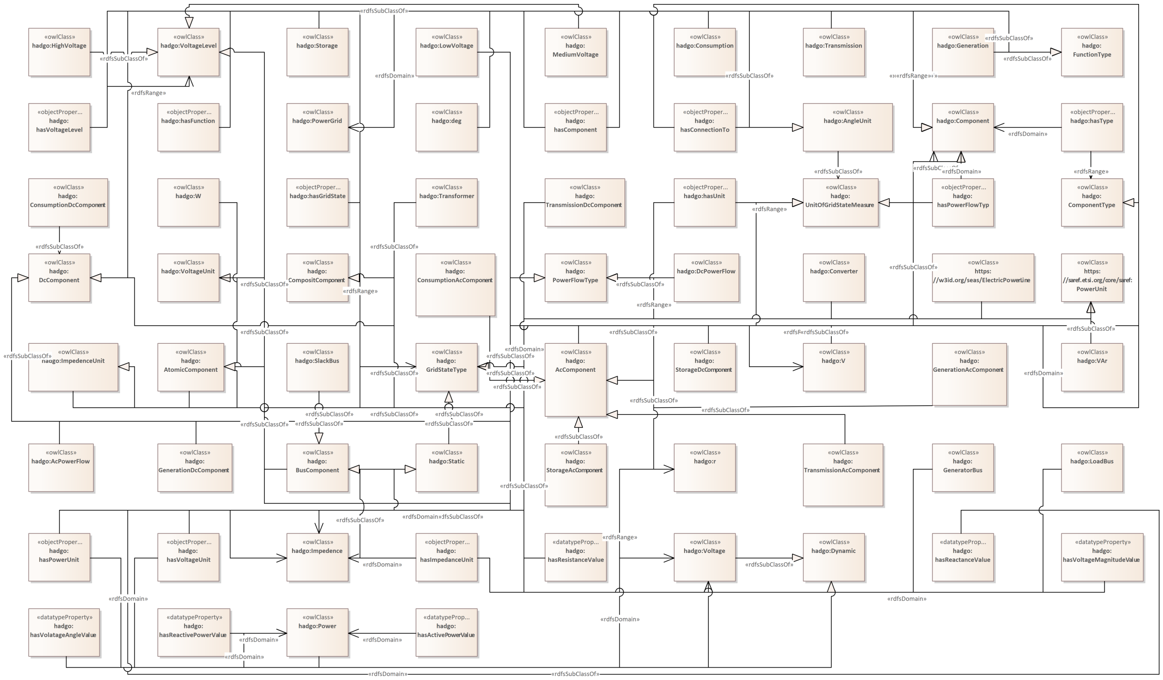

Figure 1. A partial view of the ontology with some entity classes, data, object properties, and relationships.

`hadgo:GridStateType`, `hadgo:PowerFlowType` and `hadgo:VoltageLevel`. All of these entity classes are sub-classes of `owl:Thing` entity class, which is the default way of defining classes using the Web Ontology Language (OWL).

1) PowerGrid class: The `hadgo:PowerGrid` is the umbrella entity class that can represent an AC, DC, or hybrid grid. It is also one of the top-level classes and a direct subclass of `owl:Thing`. The usage, relationships, object, and data properties for the `hadgo:PowerGrid` are summarized with the diagram shown in Figure 1. It has relations with most of the other high-level entity classes as the power grid is modeled using this class as the base. It can then include several `hadgo:Component` (and its specialized sub-classes like `hadgo:AcComponent` or `hadgo:DcComponent`). Many functional and inverse object properties help in adding more semantics to the relationship.

2) Component class: The `hadgo:Component` is one of the major entity classes in the HADGO. The `hadgo:Component` has two sub-classes for modeling an AC component `hadgo:AcComponent` and a DC component `hadgo:DcComponent`. Both AC and DC components are then divided into four sub-classes that distinguish them based on the type of function (`hadgo:FunctionType`) they are performing in the model. The asserted function types, that a component can have are defined to be either `hadgo:Generation`, `hadgo:Storage`, `hadgo:Conumption`, or `hadgo:Transmission`.

Figure 1 again can be referred to for showing the usage of the `hadgo:Component` entity class. Each `component` instance can have multiple object properties and relationships. One such relationship is with the `hadgo:ComponentType` entity class which helps in specifying the type of the specific instance of a component. A component can be either an hadgo:AtomicComponent or an `hadgo:ComponentComponent` meaning that it consists of more than one components. The `hadgo:ComponentComponent` are specialized be of two kinds `hadgo:Transformer` and `hadgo:Converter`.

Furthermore, each `hadgo:Component` instance can have object properties that assign some `hadgo:GridStateType`. These `hadgo:GridStateType` can be either `hadgo:Dynamic` or `hadgo:Static` and represents `hadgo:Voltage`, `hadgo:Power` and `hadgo:Impedance`.

Each component instance must have a relationship with `hadgo:VoltageLevel` which defines, as the name suggests, the voltage level on which the component is operating. The asserted values as the range of object property are `hadgo:HighVoltage`, `hadgo:MediumVoltage`, and `hadgo:LowVoltage`.

3) ComponentType entity class: The `hadgo:ComponentType` entity class is defined to assert the types an `hadgo:Component` can have. The two specializations for this entity class are further defined as being hadgo:AtomicComponent and `hadgo:ComponentComponent`. The former covers the component instances that are atomic and usually have a single function. In contrast, the latter covers the component instances that can be composed of more than one component. The `hadgo:ComponentComponent` are further specialized with two sub-classes `hadgo:Converter` which represents

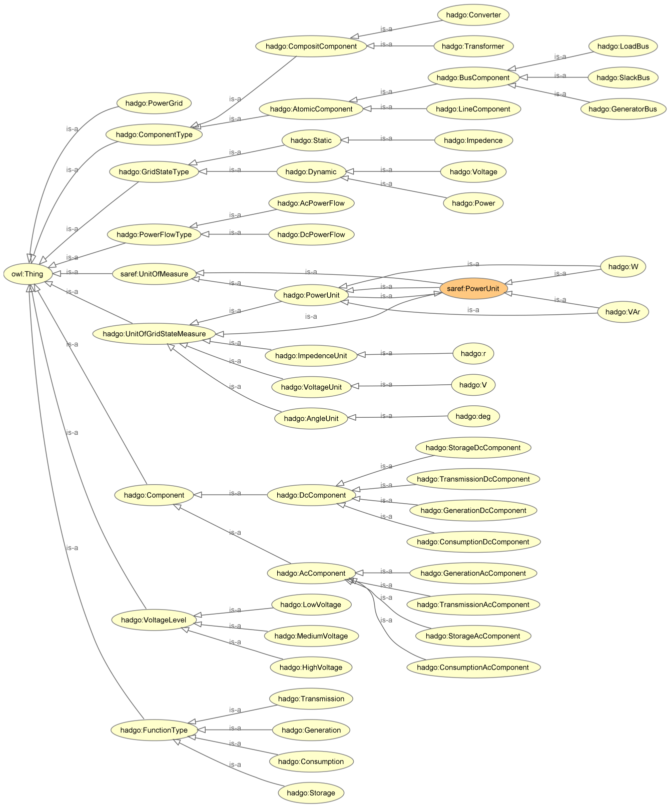

Figure 2. Hierarchy of major asserted entity classes in HADGO.

an AC/DC converter as well as `hadgo:Transfomer`.

Similarly, `hadgo:AtomicComponent` is specalized with two sub-classes that are `hadgo:BusComponent` and `hadgo:LineComponent`, which are self-explanatory. However, `hadgo:BusComponent` is further classified into three specialized sub-classes that are `hadgo:GeneratorBus`, `hadgo:LoadBus` and `hadgo:SlackBus`.

4) FunctionType class: The `hadgo:FunctionType` entity class defines the different types of function that a `hadgo:Component` can assume. The relationships and constraints are imposed using some object properties defining `hadgo:Component` as the domain and `hadgo:FunctionType` as the range. There are four specializations defined as the sub-classes that are `hadgo:Transmission`, `hadgo:Storage`, `hadgo:Generation`, `hadgo:Consumption`.

5) GridStateType class: The entity class `hadgo:GridStateType` represents the measurable grid states that an instance of a component can have. There are two sub-classes defined as `hadgo:Static` and `hadgo:Dynamic`. The two sub-classes for the `hadgo:Dynamic` are `hadgo:Voltage` and `hadgo:Power` while `hadgo:Static` only has one that is `hadgo:Impedence`.

6) PowerFlowType class: The entity class `hadgo:PowerFlowType` represents the power flows that an instance of a component in the grid can have. Changing this value affects the way this instance can be connected to other instances of the components and the type of object properties it can have. There are two specializations, `hadgo:AcPowerFlow` and `hadgo:DcPowerFlow` representing AC and DC power flow respectively.

7) VoltageLevel class: The entity class `hadgo:VoltageLevel` represents the voltage level an instance of the `hadgo:Component` can have and it defines the voltage level on which the component is operating. The three asserted sub-classes are `hadgo:HighVoltage`, `hadgo:MediumVoltage`, and `hadgo:LowVoltage`.

8) UnitOfGridStateMeasure class: The entity class `hadgo:UnitOfGridStateMeasure` represents the units that are used for measuring `hadgo:GridStatType` that a `hadgo:Component` instance can have. This entity class has children and grandchildren that define first the phenomenon and then define the respective unit.

IV. Real case scenario

The switchgear as a data instance was modeled on the HADGO ontology and validated with the HermiT Reasoner. The validated switchgear instance was provisioned on Entirety docker container on the HYPERRIDE ICT platform.

A. HADGO Switchgear Instance Modeling and Validation

The switchgear use case as an application ontology was modeled on the HADGO reference ontology in Section III above using the Protégé ontology modeling tool. The Switchgear concept was modeled as a class, and its object and data property were modeled for all its data instances.

The Hermit Reasoner was selected as the Logic evaluation solver for the switch gear data instance for the HADGO ontology and some inconsistencies during the use case modeling were observed and resolved based on the explanation results from the log output. All the instance assertions for both data and object properties were modeled and synchronized and can be used as anchor terms during ontology matching or ontology alignment which can be learned statistically or modeled and updated with Protégé.

The data instance as well as the assertions and properties defined and inferred for the Switchgear use case will also serve as a ground truth and data lineage for all future switchgear-based data analytics in the hybrid AC/DC Domain. Figure 3 below shows the modeled switch gear use case as an asserted hierarchy.

TABLE I
SUMMARY OF HADGO ASSERTED AND INFERRED FACTS ACHIEVED USING HERMIT REASONER

S.#	Fact Type	Count
1	Total Axioms achieved	301
2	Classes	63
3	Object Property	18
4	Data Property	12
5	Individual	1

The HermiT Reasoner, which is based on the Tableau Algorithm, carries out OWL Logic computation on the modeled HADGO reference ontology with the Switch Gear use case. From the decidability and satisfiability Logic computation by the HermiT Reasoner on the modeled HADGO ontology, Table I summarizes the asserted and inferred Facts validated on ontology.

These results validate the HADGO ontology as a Reference Ontology for any Hybrid AC/DC or DC grid. The Owl file in Turtle format is shown in the Appendix and can be run by anyone for verification and extension.

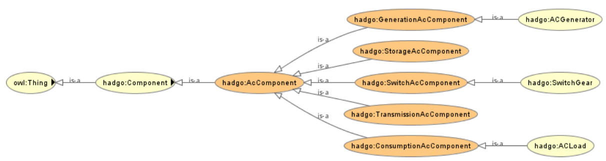

Figure 3. Hadgo Switch Gear Use Case Ontology Hierarchy.

From the HADGO base ontology, the inferred axioms data were exported and the metric was compared between the base ontology including the asserted and inferred axioms, and that of only the inferred axioms. The combined chart is shown in Figure 4 Therefore we have a total of 301 axioms in the base ontology out of which 117 are inferred axioms and 184 are asserted axioms.

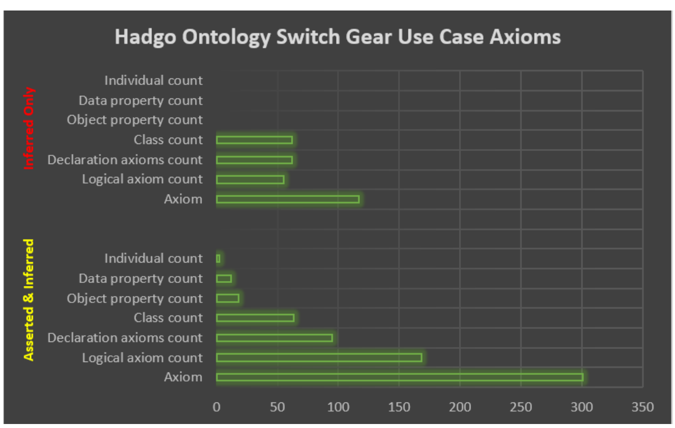

Figure 4. HADGO Ontology Switch Gear Use Case Axioms.

The achieved inference from the Switchgear use case modeling update on HADGO confirms that the logical restrictions, class constructors, class disjoint, and pairwise disjoint on multiple classes and the partitions carried out on during the HADGO ontology development were effective. It was found during the experiment that the more axioms were asserted from the domain knowledge of the AC/DC grid, the more inferred axioms were obtained.

B. HADGO Switchgear Entity Provisioning and application

The developed HADGO ontology is used in the HYPERRIDE sensing and monitoring infrastructure layer, as part of the work done for the definition of a reference ICT Platform in the AC/DC grid context [19]. The first step is to collect the sensor data through the MQTT Broker. The IoT Agent is connected with the MQTT Broker and ORION Context Broker. The Entirety [20] creates the entity of the device. The HADGO ontology is integrated into the Entirety to harmonize the data according to [21]. The data are saved in the MongoDB and the QuantunLeap subscribes to the context broker and forwards the data to the CrateDB. Finally, the data are monitored in the Grafana.

TABLE II
LIST OF COMPONENTS USED FOR TESTING THE REAL CASE SCENARIO

S.#	Component Type	Component Name
1	Main contactors	Schaltbau CT1230/08 + CT1130/08
2	Current sensors	LEM LF 1010-S current transducer
3	Voltage sensors	LEM DVM 3000 voltage transducer
4	Controller	WAGO 750 Series Modbus Bus-coupler

To test the sensing and monitoring layer, the platform is implemented on the German pilot side. The switch gear data is collected and monitored with the developed platform. The switch gear is a self-build type. The important components used are described in Table II.

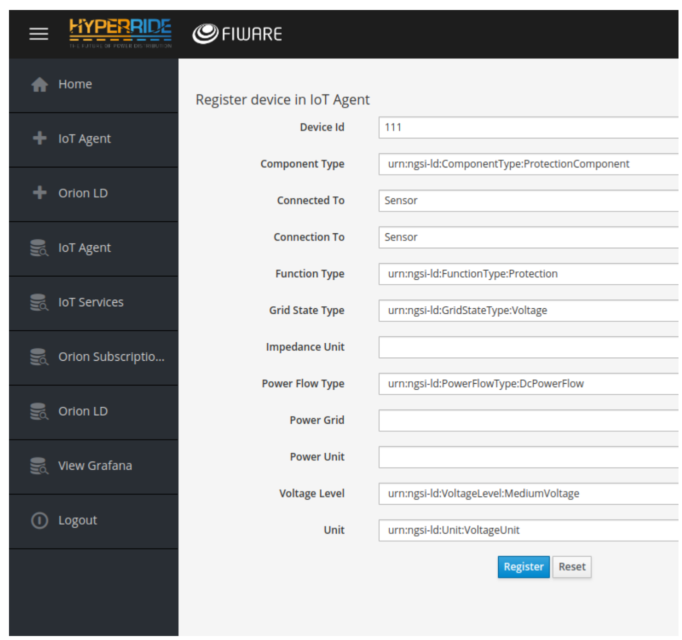

Figure 5. Register Device in IoT Agent.

The development is based on the following steps. First, the IoT agent for switch gear is created in Entirety based on the HADGO ontology. The process is shown as follows in Figure 5. The parameters of the table are defined in the data model in the code as NGSIv2, NGSI-LD, and JSON formats. The Quantum Leaps part is to read the formats of NGSIv2, NGSI-LD, and JSON. Also, the Orion Context Broker has a role here for the Format checking. The data is now uploaded to the CrateDB. Then, the data are illustrated with Grafana in Figure 6.

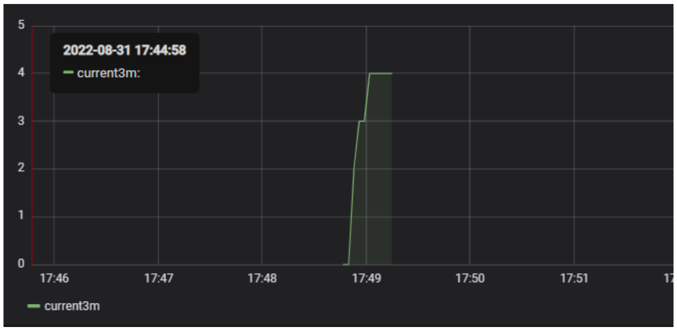

Figure 6. Monitoring the switch gear data.

V. Conclusion

The HADGO ontology is a proposed core reference ontology for the AC/DC and DC smart grid. The UML information modeling for the software applications implementation was carried out with Enterprise Architect while the ontology modeling was implemented with Protege. New AC/DC and DC grid entities with their subclasses and superclasses were developed and described for the new grid concepts. For the relationships, object properties and data properties were developed and described. For the individual, a use case of Switch Gear was described and object and data assertion were defined. The HADGO ontology was validated using the HermiT Reasoner and 301 asserted and inferred axioms were achieved in this experimental study.

For future work, more unary predicates and binary predicates axioms can be added as individuals to the HADGO ontology to increase its ground truth and axioms knowledge base.

To enable more application of the HADGO ontology at scale, future work on ontology learning can be carried out on the HADGO reference ontology using unstructured, semi-structured, and structured data from the pilot sites of hybrid AC/DC power grid.

Also, statistical methods can be carried out on the HADGO ontology for more AC/DC entity classification and relationship prediction on the modeled ontology. This helps to achieve more inferred axioms to increase the scope of data models and knowledge graphs for more data integration in information systems applications and databases.

Acknowledgment

The research leading to these results has received funding from the European Union H2020 Framework Programme (H2020/2014-2020) under Grant Agreement No 957788.

References

[1] M. Fan, H. Liang, Z. Zhang, A. Su, and W. Chen, "Cost-benefit analysis of integration der into distribution network," in *CIRED 2012 Workshop: Integration of Renewables into the Distribution Grid*, 2012, pp. 1–4.

[2] "IEEE Smart Grid Vision for Computing: 2030 and Beyond Reference Model," *IEEE*, pp. 1–18, 2016.

[3] R. Jochem and T. Knothe, "Interoperability requirements derived from interoperability dimensions," in *Enterprise Interoperability II*, R. J. Gonçalves, J. P. Müller, K. Mertins, and M. Zelm, Eds. London: Springer, 2007, pp. 39–50.

[4] N. Guarino, D. Oberle, and S. Staab, *What Is an Ontology?* Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–17. [Online]. Available: https://doi.org/10.1007/978-3-540-92673-3_0

[5] M. Ciocoiu, D. Nau, and M. Grüninger, "Ontologies for integrating engineering applications," *J. Comput. Inf. Sci. Eng.*, vol. 1, pp. 12–22, 03 2001.

[6] L. D. Lauretis, S. Costantini, and I. Letteri, "An ontology to improve the first aid service quality," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2019, pp. 1479–1483.

[7] J. S. S. Melo, T. A. Neves, and L. E. dos Santos, "Saref: Sistema de apresentação remota por reconhecimento facial," *CNJ*, vol. 6, pp. 77–92, 2022.

[8] L. Daniele, F. den Hartog, and J. Roes, "Created in close interaction with the industry: The smart appliances reference (saref) ontology," in *International Workshop Formal Ontologies Meet Industries*. Springer, 08 2015, pp. 100–112.

[9] M. de Roode, A. Fernández-Izquierdo, L. Daniele, M. Poveda-Villalón, and R. García-Castro, "Saref4inma: A saref extension for the industry and manufacturing domain," *Semantic Web Journal*, vol. 11, pp. 911–926, 2020.

[10] M. Poveda-Villalón and R. García-Castro, "Extending the SAREF ontology for building devices and topology," in *Proceedings of the 6th Linked Data in Architecture and Construction Workshop, London, United Kingdom, June 19-21, 2018*, ser. CEUR Workshop Proceedings, M. Poveda-Villalón, P. Pauwels, and A. Roxin, Eds., vol. 2159. CEUR-WS.org, 2018, pp. 16–23. [Online]. Available: https://ceur-ws.org/Vol-2159/02paper.pdf

[11] ETSI, "SAREF4ENER: an extension of SAREF for the energy domain created in collaboration with Energy@Home and EEBus associations," https://saref.etsi.org/saref4ener/v1.1.2/, 2020, [Online; accessed 2023-09-29].

[12] M. Haghgoo, I. Sychev, A. Monti, and F. Fitzek, "Sargon – smart energy domain ontology," *Smart Cities*, 12 2020.

[13] "ETSI GS CIM 009 V1.2.2 (2020-02); Context Information Management (CIM); NGSI-LD API," *ETSI Group Specification*, 2020.

[14] K. Salameh, R. Chbeir, H. Camblong, G. Tekli, and I. Vechiu, "A generic ontology-based information model for better management of microgrids," in *Artificial Intelligence Applications and Innovations*, R. Chbeir, Y. Manolopoulos, I. Maglogiannis, and R. Alhajj, Eds. Cham: Springer International Publishing, 2015, pp. 451–466.

[15] J. Ariza, F. Larrinaga, and E. Curry, "A unified semantic ontology for energy management applications," in *2nd International Workshop on Ontology Modularity, Contextuality, and Evolution (WOMoCoE 2017) - WSP/WOMoCoE@ISWC - Vienna*, 10 2017.

[16] R. Rudnicki, B. Smith, T. Malyuta, and W. Mandrick, "White Paper: Best Practices of Ontology Development," https://www.nist.gov/system/files/documents/2021/10/14/nist-ai-rfi-cubrc_inc_002.pdf, Oct 25 2016, [Online; accessed 2023-04-24].

[17] B. Smith and W. Ceusters, "Ontological realism: A methodology for coordinated evolution of scientific ontologies," *Appl Ontol.*, vol. 5(3-4), pp. 139–188, 2010.

[18] Y. Alfaifi, "Ontology development methodology: A systematic review and case study," in *2022 2nd International Conference on Computing and Information Technology (ICCIT)*, 2022, pp. 446–450.

[19] M. Mammina, A. Rossi, D. Arnone, G. Zizzo, S. Moradi, and P. Smith, "Predictions in Resilient Hybrid AC/DC Grids Leveraged by an Interoperable and Secure ICT Platform," in *2022 IEEE International Conference on Environment and Electrical Engineering and 2022 IEEE Industrial and Commercial Power Systems Europe (EEEIC / ICPS Europe)*, 2022, pp. 1–6.

[20] M. Haghgoo, I. Sychev, A. Monti, and F. H. Fitzek, "Entirety—semanntic provisioning and governing iot devices in smart energy domain," *SoftwareX*, vol. 18, p. 101081, 2022.

[21] Z. Pan, G. Pan, and A. Monti, "Semantic-similarity-based schema matching for management of building energy data," *Energies*, vol. 15, no. 23, p. 8894, 2022.