Web Semantics: Science, Services and Agents on the World Wide Web 61–62 (2020) 100550

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

DABGEO: A reusable and usable global energy ontology for the energy domain

Javier Cuenca [a,*], Felix Larrinaga [a], Edward Curry [b]

[a] Mondragon University/Faculty of Engineering, Loramendi 4, 20500 Arrasate-Mondragon, Spain
[b] Insight Centre For Data Analytics, National University of Ireland, Galway, IDA Business Park, Lower Dangan, Galway, Ireland

A R T I C L E   I N F O

Article history:
Received 16 April 2019
Received in revised form 29 November 2019
Accepted 27 January 2020
Available online 5 February 2020

Keywords:
Ontology
Energy domain
Ontology reusability
Ontology usability

A B S T R A C T

The heterogeneity of energy ontologies hinders the interoperability between ontology-based energy management applications to perform a large-scale energy management. Thus, there is the need for a global ontology that provides common vocabularies to represent the energy subdomains. A global energy ontology must provide a balance of reusability–usability to moderate the effort required to reuse it in different applications. This paper presents DABGEO: a reusable and usable global ontology for the energy domain that provides a common representation of energy domains represented by existing energy ontologies. DABGEO can be reused by ontology engineers to develop ontologies for specific energy management applications. In contrast to previous global energy ontologies, it follows a layered structure to provide a balance of reusability–usability. In this work, we provide an overview of the structure of DABGEO and we explain how to reuse it in a particular application case. In addition, the paper includes an evaluation of DABGEO to demonstrate that it provides a balance of reusability–usability.

1. Introduction

Energy management deals with monitoring and controlling the energy usage with different objectives, i.e., improve the energy efficiency, reduce the energy cost [1]. Energy management of current infrastructures is evolving towards the future Smart Grid. The Smart grid is envisioned as the next generation power grid. It integrates Information and Communication Technologies (ICTs) to the existing power grid. With the integration of ICTs, the Smart Grid aims to improve current grid efficiency and sustainability by integrating ICT-based energy management applications (also known as *Smart Grid energy management applications* [2]). Smart Grid energy management applications (1) optimize the use of both non-renewable and renewable energy sources, (2) suggest citizens actions to change their energy management behavioural patterns for economic, social and ecological purposes and (3) collaborate with humans to prevent and react to power outages caused by power peak periods or natural disasters [2]. To achieve these objectives, energy management applications must interact with humans. They must also collect, exchange and extract knowledge from data from heterogeneous and complex energy data domains at high rates and in real-time. We consider a data domain as a set of related concepts that belong to a specific area of interest [3]. The energy data domains include energy performance data (i.e., energy consumption, renewable energy production) and energy-related contextual data (i.e., infrastructure data, weather data) [4].

Current research in energy management evidences the development of energy ontologies to meet these challenges, i.e., ThinkHome ontology [5], SAREF4EE ontology [6], ProSGv3 ontology [7], SEMANCO ontology [4] and EnergyUse ontology [8]. These ontologies represent semantically different energy data domains. Semantically represented energy knowledge improves the performance of intelligent agents and data analysis applications used for knowledge extraction and decision-making within energy management applications [2]. Hence, this knowledge is used as a knowledge base by Smart Grid energy management applications. These applications are deployed in specific Smart Grid scenarios (infrastructures of the Smart Grid such as smart homes or buildings). Therefore, the Smart Grid energy management applications can be classified into the following types depending on the Smart Grid scenarios where they are deployed [2] (we consider an *application type* a family of applications that perform similar tasks or have similar objectives [9]): smart home energy management applications (focused on controlling and monitoring home device energy operation), building/district/city energy management applications (focused on giving a complete energy performance assessment of buildings and districts), organization

* Corresponding author.
E-mail addresses: jcuenca@mondragon.edu (J. Cuenca), flarrinaga@mondragon.edu (F. Larrinaga), edward.curry@insight-centre.org (E. Curry).

https://doi.org/10.1016/j.websem.2020.100550
1570-8268/

energy management applications (focused on providing a holistic view of organization energy performance) and Smart Grid Demand Response (DR) management applications (focused on managing the energy consumption of infrastructures in response to the current energy supply conditions [1]). We define these application types as *Smart Grid scenarios* [2]. The applications of each Smart Grid scenario can be classified into more specific application types, since they have specific objectives in common and perform highly related tasks. Therefore, each Smart Grid scenario encompasses more specific application types. For instance, within smart home energy management applications, there are applications focused on home energy assessment, home energy saving advice and home appliances DR management. For more detail about these energy management application types, we refer the reader to [2].

1.1. Motivation

The representation of many energy data domains is repeated across the existing energy ontologies. However, they represent the knowledge for the same domains applying different vocabularies, leading to heterogeneous ontologies and knowledge bases [2]. Current ontology-based energy management applications are limited to pilot demonstrators deployed in specific Smart Grid scenarios (i.e., homes, buildings). To improve current grid efficiency and sustainability at a greater scale, ontology-based energy management applications that operate in different infrastructures will be required to exchange knowledge. This knowledge exchange would be hampered by the heterogeneity of energy ontologies. Hence, there is the need to create a *global* or *standard* ontology that provides a common knowledge representation of energy domains [2]. Global ontologies are ontologies that include common vocabularies to provide a common representation and a shared understanding of the domain [10]. The common knowledge of a global energy ontology can be reused to develop ontologies for energy management applications deployed in different scenarios, thus leading to interoperable knowledge bases [11].

Although a global energy ontology intends to unify the energy domain representation, each application has individual requirements. An ontology cannot represent all the knowledge required by any application that can reuse it. Therefore, we must assume that a global ontology cannot be reused in each application without adapting it to the application requirements [12]. These adaptation activities are known as ontology reengineering activities [13], which include knowledge extension and modification. In particular, the ontology may be extended to include the specific knowledge required by the application [14]. The knowledge that does not cover the application ontology requirements or the one not needed to exchange information with other applications may also be discarded (i.e., so that this knowledge does not affect the computation performance of the application that processes the ontology knowledge). This activity is known as ontology pruning [13]. These changes affect to the specific knowledge required by each application. Therefore, the knowledge that the different applications have in common would still be represented with the same vocabularies, thus enabling interoperability [9]. However, even if the same concepts and properties are reused in different applications, their meaning might have been completely changed by performing ontology reengineering activities (i.e., by removing axioms that describe a certain concept). Hence, this limitation should be taking into account when exchanging knowledge between different applications that reuse the global ontology.

Taking this into account, an ontology reused in different applications (which is the case of a global energy ontology) must minimize the ontology reuse effort so that it can be reused by ontology developers in different applications [14].

On the one hand, the ontology must be *reusable* [9,14]. Ontology *reusability* was defined by Pâslaru-Bontas [12] as "*the adaptation capability of an ontology to arbitrary application contexts*". Nevertheless, it is not feasible to develop an ontology that is appropriate for all application contexts. Rather, a reusable ontology must support a set of applications in a given domain and must be easily adaptable [15]. To provide reusability, the ontology must include the abstract domain knowledge reused by many applications. However, if the ontology is too abstract, the effort of extending its knowledge to satisfy specific requirements will be high. Thus, ontology developers are less likely to reuse the ontology to develop ontologies for their applications.

Considering this, the ontology must be also *usable* [9,14]. Ontology *usability* deals with reducing the effort required to adapt the ontology so that it can be used in a given application context [12]. A usable ontology minimizes the ontology reuse effort when it is reused to develop ontologies for specific applications. To provide usability, the knowledge of the ontology must be as specific as possible to ease its adaptation to specific application requirements. Nevertheless, if the ontology represents the knowledge required by a specific application, the effort of adapting the ontology to other applications with different knowledge requirements would be high [9,14]. In particular, the knowledge required by other applications should be added and the knowledge not required by these applications may be discarded.

Both ontology reusability and usability are objectives "*in natural conflict*" [14]. Hence, an ontology that supports different applications must achieve a balance between reusability and usability so that it can be reused in different applications with moderate effort [9,14]. This challenge is known as the ontology *reusability–usability tradeoff problem* [16]. Achieving a balance of reusability–usability is particularly important in extensive and complex domains, since these domains will require large-scale ontologies.

Since the energy domains are complex, a global energy ontology will be a large-scale ontology reused in different applications. Therefore, it should provide a balance between reusability and usability.

To date, *layered ontologies* have been applied to achieve a balance of reusability–usability [17]. They classify into different abstraction layers the *common domain knowledge* (abstract knowledge reused by most applications) and the *variant domain knowledge* (specific knowledge reused only by certain application types) [14,18]. In addition, the knowledge of each layer is divided into small ontologies known as ontology modules [19]. Therefore, we can consider layered ontologies as a kind of ontology networks (ontologies that are made up by interconnected ontology modules [13]) that classify the domain knowledge into different levels of abstraction.

This layered structure (along with ontology modularization) enables ontology developers to reuse only the necessary knowledge to develop ontologies that satisfy the knowledge requirements of their applications. Therefore, the number of activities needed to adapt the ontology to different application requirements is reduced, thus reducing the ontology reuse effort in different applications [15].

1.2. Contribution

As our main contribution, this paper presents and describes the v1.0 of DABGEO (Domain Analysis-Based Global Energy Ontology), a reusable and usable global ontology for the energy domain. DABGEO is a large-scale ontology that includes 97 modules. The modules of DABGEO are published and can be downloaded at the DABGEO home page: http://www.purl.org/dabgeo.

The ontology is licensed under the Creative Commons Attribution 4.0.[1] DABGEO provides a common representation of the energy domains represented heterogeneously by the available energy ontologies developed for specific applications. The vocabularies of DABGEO can be reused by ontology engineers to develop ontologies for specific energy management applications.

In contrast with previous global energy ontologies, the main contribution of DABGEO is that it classifies the common domain knowledge and variant domain knowledge into different abstraction layers. With this structure, DABGEO provides a balance between reusability–usability to reduce the ontology reuse effort in different applications.

The paper also presents an evaluation of DABGEO to demonstrate its balance of reusability–usability. In particular, we examined how two ontology engineers reused DABGEO in two energy management applications. The reuse effort of DABGEO was compared with the effort of reusing a previously developed global energy ontology.

The paper is structured as follows: in Section 2, DABGEO is positioned with respect to previously developed energy ontologies to highlight its main contributions. Section 3 summarizes the design and development principles of DABGEO. Section 4 describes the content and structure of DABGEO. Section 5 shows an example of DABGEO usage in a specific application case. Section 6 presents an empirical evaluation for DABGEO conducted to determine its balance of reusability–usability. Section 7 discusses the ontology evaluation results. Finally, Section 8 summarizes the conclusions of the study as well as future lines of work.

2. Related work

This section provides an overview of a set of energy ontologies developed in the last decade and positions DABGEO with respect to them. The overview includes the ontologies that support specific energy management applications, since DABGEO was developed to provide a common representation of these ontologies. The overview also includes previously developed global energy ontologies.

2.1. Ontologies developed for specific energy management applications

From the beginning of the current decade, energy ontologies for applications that operate in different Smart Grid scenarios and that have different purposes have been developed.

Kofler et al. [5], Daniele et al. [6], and Burel et al. [8] presented ontologies for smart home energy management applications. Kofler et al. [5] presented the ThinkHome ontology, which is expected to be used to represent the knowledge bases of multi-agent smart home energy management systems. Daniele et al. [6] presented the ontology SAREF4EE. The objective of SAREF4EE is to improve interoperability among electrical appliances of different manufacturers allowing them to be connected with customer energy management systems used for Smart Grid DR optimization strategies. Burel et al. [8] developed the EnergyUse ontology, used to create the knowledge base of a collaborative web platform that aims to raise awareness for home end-users of climate change.

Curry et al. [20], Stavropoulos et al. [21] and Blomqvist and Thollander [22] presented building, facility and organization energy data representation ontologies. Curry et al. [20] developed ontologies to represent and link enterprise knowledge. Stavropoulos et al. [21] presented the BonSai ontology, which supports a building energy management system that monitors the energy performance and allows users to take actions to increment energy savings. Blomqvist and Thollander [22] developed an ontology that represents the knowledge about energy efficiency improvements, energy saving recommendations and energy measures taken from previous energy audits.

Gillani et al. [7] and Corrado et al. [4] presented ontologies for other Smart Grid scenarios. Gillani et al. [7] presented the ProSGv3 ontology, which represents the energy data of prosumer oriented Smart Grids. Corrado et al. [4] presented the SEMANCO ontology to provide models for urban energy systems that assess the energy performance of urban areas.

Considering the purpose of the reviewed ontologies and the applications where they are reused, Table 1 summarizes the Smart Grid scenario and the application types supported by each ontology. For more detail about this classification we refer the reader to [2].

Each ontology was developed to support the applications that operate in a certain Smart Grid scenario (i.e., smart homes, buildings, districts). Some ontologies support the applications of the same scenario. For instance, ThinkHome and EnergyUse were developed to be reused in smart home energy management applications. However, each ontology is reused by applications with different purposes. For example, ThinkHome was developed to be reused in applications focused on home energy assessment and device control, while EnergyUse is reused in applications focused on giving advice on energy saving.

In contrast to these ontologies, DABGEO aims to be a more general-purpose ontology that provides a common representation of the energy domains they represent. DABGEO can be reused to develop ontologies for management applications that operate in different scenarios. The knowledge that these applications have in common would be represented with the same vocabularies. Therefore, the knowledge exchange between different applications would be enabled [2].

2.2. Global energy ontologies

In recent years, several global energy ontologies that enable interoperability between different ontology-based energy management applications have also been developed.

On the one hand, the authors developed the OEMA ontology network [23], which provides a common representation of the energy domains represented heterogeneously by the energy ontologies reviewed in Section 2.1. To see in more detail how the heterogeneous vocabularies of existing ontologies were integrated into OEMA, we refer the reader to [23]. OEMA is divided into several ontologies that represent one energy domain each, including the common and variant knowledge. OEMA puts emphasis on being detailed and complete, even at the cost of being less reusable and usable. Therefore, ontology developers must extract and adapt the knowledge they need from the OEMA ontologies each time they develop an application ontology. DABGEO covers the same energy domains as OEMA. In contrast, DABGEO separates the common and variant domain knowledge into abstraction layers to enable ontology developers to reuse the necessary knowledge when developing application ontologies. Hence, DABGEO can be seen as an improved version of the OEMA ontology network.

On the other hand, Lefrançois [24] presented the SEAS ontology. SEAS is a modular ontology that represents different energy domains to enable interoperability between smart systems that manage the operation of the future energy grid. SEAS represents the abstract domain knowledge reused by many applications (i.e., it includes concepts such as *Device* or *Observation*), thus enabling ontology reuse in different applications. Therefore, depending on the application where it is reused SEAS may require a significant effort to extend its knowledge to satisfy specific

1 https://creativecommons.org/licenses/by/4.0/

Table 1
Smart Grid scenarios and application types supported by energy ontologies.

Smart Grid scenario	Application type	Ontology
Smart home energy management applications	Home energy assessment and device control applications	ThinkHome ontology [5]
	Home energy saving advice applications	EnergyUse ontology [8]
	Home appliances Demand Response management applications	SAREF4EE ontology [6]
Building/district/city energy management applications	City energy performance assessment applications	SEMANCO ontology [4]
	Building energy saving advice applications	BonSAI ontology [21]
Organization energy management applications	Organization energy saving advice applications	Ontology developed by Blomqvist and Thollander [22]
	Organization energy assessment applications	Ontologies developed by Curry et al. [20]
Smart Grid demand response management applications	–	ProSGv3 ontology [7]

knowledge requirements. In addition to including abstract domain knowledge, DABGEO includes specific knowledge reused only by certain application types (i.e. knowledge about devices used in smart home energy management applications). Hence, in contrast to SEAS, DABGEO enables ontology developers to reuse both abstract knowledge and specific knowledge that is closer to application requirements.

3. Design and development principles

The main objective of DABGEO is to provide a balance of reusability–usability to cover the gaps of existing global energy ontologies. Hence, it follows the design principles of previous reusable and usable ontology approaches. In particular, DABGEO resembles the structure and follows design principles from OntoCape [14] a well-known reusable and usable ontology developed for the chemical process engineering domain:

- *Abstraction layering:* the ontology classifies the common and variant domain knowledge into different abstraction layers.
- *Loosely coupled and self-contained ontology modularization:* the knowledge of each layer is divided into ontology modules that represent closely related topics. The boundaries of the ontologies are chosen so that the modules are independent. That is, so that they only relate with the modules whose knowledge they extend or depend on.

In addition, DABGEO must provide a common energy domain representation to enable the development of interoperable knowledge bases. Considering this, DABGEO was developed taking as a starting point the OEMA ontology Network[2] (introduced in Section 2.2). The design and development team of DABGEO included people with expertise in the energy domain and ontology engineers.

The development process of DABGEO begun with the definition of the knowledge of OEMA that should be included in each layer by domain experts and ontology engineers. This knowledge classification was performed based on the knowledge similarities and differences of existing energy domain ontologies. The knowledge relevant to the domain and the knowledge reused by most ontologies was considered as common, since it is reused by most of energy management applications. The knowledge required by specific application types and reused in specific ontologies was considered as variant. To see in more detail how the layered structure of DABGEO was designed, we refer the reader to [17].

Then, the knowledge of OEMA was partitioned into ontology modules that represent closely related topics. The modules were placed into each layer according to the knowledge classification performed. Only the properties of OEMA used to express high-level relations between the knowledge from different energy data domains [23] were not included in DABGEO. Therefore, with the exception of the aforementioned properties, the knowledge from DABGEO is semantically equivalent to the knowledge from OEMA [19].

In conclusion, we can consider DABGEO as a second and improved version from OEMA.

4. Overview of the DABGEO ontology

This section describes the content and structure of DABGEO, as well as the main benefits of this structure. DABGEO includes 97 modules, which were implemented in OWL-2 DL [25] with the Protégé ontology editor. Concepts, relations, and attributes were modelled as classes, object properties and data properties, respectively. In total, DABGEO includes 1965 classes, 276 object properties and 198 data properties. Axioms were represented in Protégé using various kinds of OWL restrictions (i.e., cardinality restrictions, object property restriction or datatype restrictions).

Since DABGEO is a large-scale ontology, describing the main classes and properties of each module would make it difficult to understand the structure of DABGEO. Therefore, this section offers a high-level description of DABGEO content and structure without going into detail in each module. The detailed description and specification of each module can be found at the DABGEO home page.[3]

4.1. DABGEO content

DABGEO covers the following energy data domains:

- **Energy equipment domain:** knowledge about energy equipment and energy device performance. The main classes of DABGEO used to represent the main concepts of this domain are the following: ***Device*** (used to represent different types of devices), ***EnergyConsumptionSystem*** (used to represent energy consumption devices such as home appliances, Heating Ventilation and Air-Conditioning (HVAC) systems), ***EnergyGenerator*** (used to represent energy generation devices such as solar panels) and ***MeteringActuation***

2 http://www.purl.org/oema

3 http://www.purl.org/dabgeo

(used to represent sensors and actuators). The knowledge about these concepts is extended by classes, properties and axioms used to represent the knowledge about energy equipment features and operational aspects such as device energy consumption or device power profile.
- **Infrastructure domain:** knowledge about infrastructures and buildings. The main class of DABGEO used to represent the knowledge about this domain is the ***Infrastructure*** class. This class is used to represent different types of infrastructures such as homes, buildings or power plants. The knowledge about these concepts is extended by classes, properties and axioms used to represent the knowledge about building/infrastructure features (i.e., surface, material), geometrical details (rooms, floors) and internal and external environmental conditions (i.e., room temperature).
- **Energy performance data domain:** knowledge about the energy performance of devices and infrastructures. The main class of DABGEO used to represent the knowledge about this domain is the ***EnergyParameter*** class. This class is used to represent energy performance values such production, consumption and storage values and energy key performance indicators such as infrastructure energy cost or energy gain. The knowledge about these concepts is extended by classes, properties and axioms used to represent specific energy performance values and indicators.
- **Energy external factors domain:** knowledge about external factors that may affect the energy usage, i.e., weather conditions, socio-economic data. The main classes of DABGEO used to represent the main concepts of this domain are the following: ***WeatherPhenomenon*** (used to represent weather conditions)***,*** **SocioEconomicFactor** (used to represent the basic overall social and economic data pertaining to the population) and ***EnvironmentalFactor*** (used to represent the principal air pollutants in the urban area).
- **Smart Grid stakeholders domain:** knowledge about energy stakeholders. The main classes of DABGEO used to represent the main concepts of this domain are the following: ***Actor*** (used to represent actors that participate in the usage process such as home users, building occupants and organizations) and ***EnergyMarketRole*** (used to represent roles that energy actors have in the energy market, such as energy consumers or producers). The knowledge about these concepts is extended by classes, properties and axioms used to represent the following knowledge: actor preferences about energy devices (i.e., minimum/maximum price that the user is willing to pay for energy production/consumption), organization internal structure (i.e., organization members and business processes) or the energy type provided by the energy providers (i.e., electric energy, thermal energy).

4.2. DABGEO structure

The structure of DABGEO is explained using as an example the *energy equipment domain* to simplify the explanation. We refer to the DABGEO home page for further information about the representation of the rest of the domains.

Within DABGEO, the energy domains are divided into subdomains that cover the knowledge of important parts of the domain. Fig. 1 provides and overview of the high-level structure of DABGEO, enumerating the subdomains in which DABGEO domains are classified. In particular, the *energy equipment domain* encompasses the following subdomains:

- **Energy consumption systems subdomain:** specific energy consumption devices such as HVAC systems, appliances, lighting systems and security systems. It also includes operational aspects of these devices such as appliance working modes.
- **Distributed energy sources subdomain:** specific energy generation systems and energy storage systems, as well as their operational aspects.
- **Metering/actuation equipment subdomain:** different types of sensors and actuators (i.e., environmental sensors, building element sensors/actuators). It also includes configuration and operational data about these devices.
- **Device operation data subdomain:** operational aspects of any device such as device commands, device functionality, device state or device power profile.

Fig. 2 provides a detailed overview of the structure of DABGEO concerning the *energy equipment domain*. The knowledge of the subdomains is divided into ontology modules that represent the knowledge of a particular topic of the subdomain (to simplify the understanding of Fig. 2, we have omitted a couple of modules and module relationships). The modules of DABGEO are classified into three abstraction layers. In the next subsections, we describe the kind of knowledge included in each layer (the list of all the modules included in each layer can be found at the DABGEO home page).

4.2.1. Common-domain layer

The *common-domain layer* includes the domain knowledge common to all Smart Grid scenarios introduced in Section 1. For instance, the *Device ontology module* represents the *Device* concept and device main properties (i.e., device name). As another example, the *energy consumption systems ontology module* represents the knowledge about energy consumption system types, i.e., knowledge about appliances or HVAC systems. This module extends the knowledge about devices, so it imports the knowledge of the *device ontology module*.

Since the knowledge of this layer is common to all Smart Grid scenarios, the modules include abstract concepts and relations of the domain (i.e., device, appliance). Therefore, the expressivity of the modules of this layer is lower than in the rest of layers [14]. For instance, the expressivity of the *Device* and *energy consumption systems* ontology modules is the *ALH(D)* description logic. This indicates that these modules are basically made up class hierarchies and properties and that they have limited reasoning potential [26].

4.2.2. Variant-domain layer

The *variant-domain layer* includes the variant domain knowledge still common to more than one Smart Grid scenario.

The knowledge of this layer is relevant to fewer applications. The modules of this layer extend and import the knowledge of the *common-domain layer,* since they include more specific concepts, relations and axioms. For example, in the *energy consumption systems subdomain* these modules represent the knowledge about specific appliances or HVAC systems, such as *general use brown goods* (i.e., body care devices) or *air conditioning systems* (i.e., space cooling systems). Therefore, the modules of this layer need more expressivity with respect to the modules of the *common-domain layer* [14]. For instance, the expressivity of the *device state ontology module,* which represents the knowledge about device state types (i.e., continuous state), is the *ALCHIQ(D)* description logic. This means that this module, apart from including classes and properties, includes cardinality restrictions over them [26].

4.2.3. Domain-task layer

The *domain-task layer* includes the domain knowledge reused in specific Smart Grid scenarios. This layer is divided into two sublayers: *scenario sublayer* and *application type sublayer*. These sublayers separate the knowledge reused only by a specific application type from the knowledge still relevant to all the application types encompassed by the Smart Grid scenario. In these sublayers, the modules are classified according to the Smart Grid scenario or application type that reuse them:

- The *scenario sublayer* represents the knowledge relevant to a certain Smart Grid scenario. For example, the knowledge about *device commands* or *device functionality* (represented by the homonymous modules) is only relevant to smart home energy management applications. As another example, knowledge about *district energy generation systems* is only relevant to different types of building/district/city energy management applications (see Fig. 2).
- The *application type sublayer* represents the knowledge reused only by certain energy management application types from a specific Smart Grid scenario. For example, within smart home energy management applications, only home appliances DR management applications reuse the knowledge about appliance operation such as *device power profile* or *appliance working modes* (represented by the homonymous modules, see Fig. 2).

The modules of the *domain-task layer* represent specific knowledge reused in specific energy management application types. Therefore, the modules of this layer specialize the knowledge from previous layers (including specific concepts, relations and axioms required by specific applications) and they use a more expressive language [14]. For example, the expressivity of the *device power profile* ontology module is the *SROIQ(D)* description logic. This indicates that the module adds more complex restrictions to class hierarchies and properties (i.e., disjoint relations, object value restrictions, inverse properties) to increase the reasoning potential [26].

4.3. DABGEO main benefits

The layered structure followed by DABGEO provides the following benefits when reusing the ontology to achieve a balance between reusability–usability:

1. *Selection of domain knowledge at the proper level of abstraction* [15]: ontology developers can analyse and select at the proper level of generality and abstraction only the necessary knowledge to develop application ontologies. Depending on the application developed, ontology developers can just use and adapt modules that include abstract knowledge or modules that include both abstract and specific knowledge. For example, a home energy management application and a district energy management application may require different specific knowledge and thus they may reuse different modules from the *domain-task layer*. In contrast, these applications may share the knowledge from upper layers.
   In addition, DABGEO classifies the variant domain knowledge according to the application types that reuse it. This feature enables ontology developers to focus on analysing and reusing the modules that contain the knowledge reused by similar applications to the one they must develop. For example, let us consider an ontology developer who reuses DABGEO to develop an ontology for an application that manages the home appliances energy consumption to adjust it to energy tariffs. Considering the goal of the application, it can be considered as a smart home energy management application [2]. Therefore, the ontology developer can focus on analysing and reusing the modules of the *domain-task layer* that are classified into smart home energy management applications.
   In conclusion, the number of activities needed to adapt the ontology to different application requirements is reduced, thus reducing the ontology reuse effort in different applications [15].
2. *Understandability and adaptability* [15]: the division of the knowledge of each layer into ontology modules reduces the complexity and facilitates the ontology understanding with respect to an ontology that represents all the knowledge in a single ontology. Since the modules are independent, they can be reused, adapted and combined to develop application ontologies without affecting other parts of the ontology [19,27].

5. DABGEO usage

This section shows an example of the usage of DABGEO in a specific application case.

So far, DABGEO has been reused in two energy management systems developed within the Rennovates European project[4]: (1) a Green Energy Provider Selection System (GEPSS) that provides the home user with a list of the available green energy providers and (2) an infrastructure Energy Performance Assessment System (EPAS) that provides a holistic view of the energy generation performance of green buildings self-sufficient in solar energy. Both systems were integrated into a pilot demonstrator of the Rennovates project deployed in Mondragon University, Abadiano municipality and Urkiola natural park (Spain).

To explain the usage of DABGEO in this section, we take as an example the GEPSS. In the following subsections, we describe the architecture of the GEPSS and we explain how DABGEO was reused in this system.

5.1. GEPSS description

The GEPSS is a multi-agent system that provides to the home energy consumer a list of the available green energy providers in the area where the home is located. The available energy providers are the ones that have surplus energy. Specifically, the system displays the provided energy type (i.e., electric energy, thermal energy), the energy source (i.e., solar power, wind power), the infrastructure that generates the energy (i.e., a solar panel installation) and the price at which the supplier sells the energy. The GEPSS data is represented through an application ontology and stored in a semantic repository used as knowledge base by the system.

Fig. 3 shows the GEPSS architecture and operation. Within the demonstrator, solar panels and batteries are installed in three infrastructures: a sports centre building, a playground and a water deposit. The energy production and battery charging state of each infrastructure is measured and an embedded system dumps these data in real-time into a database deployed in Mondragon University servers. The GEPSS is also deployed in these servers and includes three main elements: the semantic converter, semantic repository and the user interface. The semantic converter converts the data from the database into semantically represented data (according to the GEPSS ontology vocabulary) and stores it in the semantic repository. The repository also includes static data about the energy providers, infrastructures, the energy source/type they produce and the energy tariff that energy providers assign to each energy source. Finally, the user interface queries the energy data stored in the repository and displays the information about the available green energy providers. In this demonstrator, we consider that energy providers are available when the battery attached to their infrastructures is at its maximum capacity.

4 https://rennovates.eu/

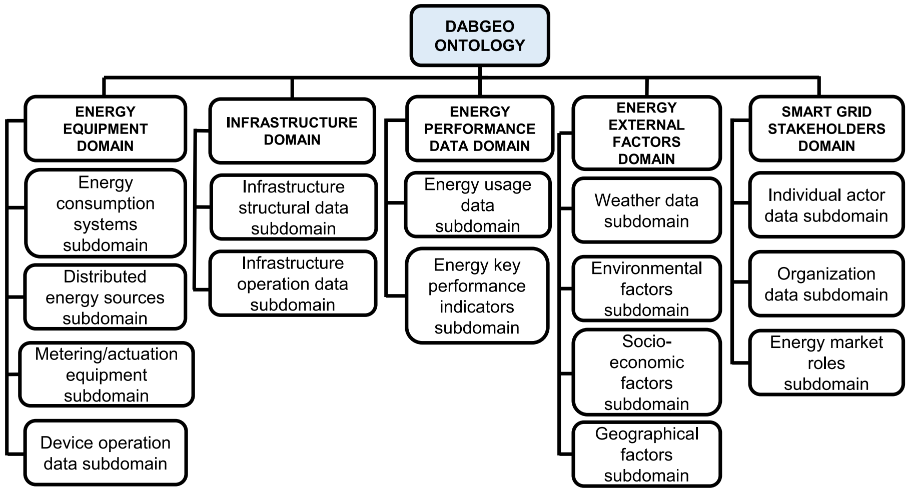

Fig. 1. High-level structure of the DABGEO ontology.

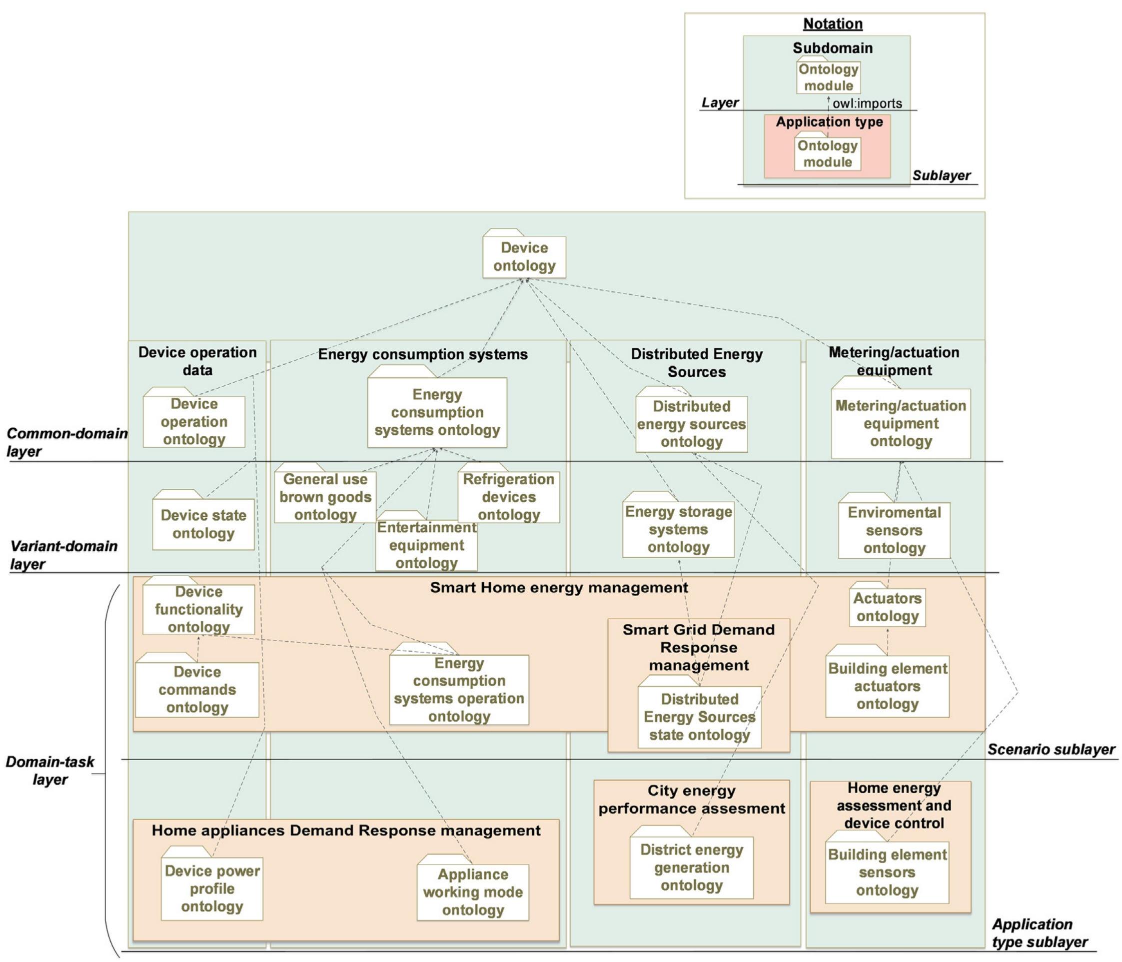

Fig. 2. Structure of the DABGEO ontology (energy equipment domain).

5.2. Development of the GEPSS ontology reusing DABGEO

The modules of DABGEO were reused to develop the ontology of the GEPSS. The GEPSS ontology was developed by ontology engineers (in collaboration with domain experts) and it was implemented with Protégé.

First, the functional ontology requirements of the GEPSS were defined by the ontology engineers in collaboration with domain experts. The requirements were defined as a set of competency questions (CQs) [28], that the GEPSS ontology must answer. In the GEPSS use case, the CQs correspond to the queries the user interface makes to the semantic repository. As an example, below we list some of the defined CQs:

– **CQ1:** What is the name of an infrastructure?

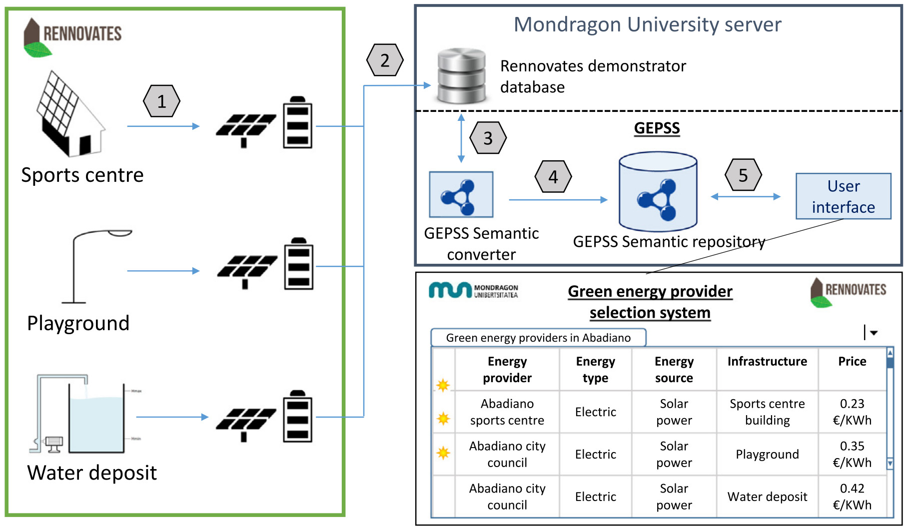

Fig. 3. Architecture of the GEPSS.

– **CQ2:** Which are the infrastructures owned by an energy provider?
– **CQ3:** What is the type of energy (i.e., heating, electricity) provided by an energy provider?

Second, a set of terms and relations that the ontology must represent to answer the defined CQs were extracted. For instance, to answer the aforementioned CQs the GEPSS ontology must include concepts such as *EnergyProvider, EnergyType, EnergyGenerationSystem or Infrastructure*. In addition, the GEPSS ontology must include relations such as *EnergyProvider providesEnergyType EnergyType*.

Third, the description of DABGEO modules was analysed in the DABGEO home page to identify those modules that may contain the concepts and relations that the GEPSS ontology must include. Then, the specification of the identified ontology modules was analysed to check the requirements they solve. In addition, ontology engineers opened in Protégé the identified modules to search for the ontology elements that represent the terms and relations the GEPSS must include to answer the defined CQs. Those modules that include the necessary elements to solve part of the requirements of the GEPSS ontology were selected for reuse.

The ontology engineers started analysing the modules from the *domain-task layer*, since the knowledge from these modules is closer to application requirements. In particular, they analysed the modules reused by *smart home energy management applications*, since the GEPSS is expected to be used at homes. For instance, among the modules reused by smart home energy management applications, the *provided energy type ontology module*[5] of DABGEO includes the knowledge about the energy types provided by energy providers. Fig. 4 shows the description and part of the specification of this module. Going deeper into its specification, the *provided energy type ontology module* answers the CQ3 of the GEPSS ontology requirements: *What is the type of energy provided by an energy provider?*. This module was opened in Protégé to search for the elements that the GEPSS ontology must represent to answer the CQ3. In particular, this module includes the ***EnergyProvider*** and ***EnergyType*** classes, which are used to represent energy providers and the energy type they provide respectively. In addition, it includes the ***providesEnergyType*** property to indicate the energy type provided by an energy provider. Considering this, the *provided energy type ontology module* satisfied part of the requirements of the GEPSS ontology and was selected for reuse. Then, the ontology engineers continued analysing the modules from upper layers to find the knowledge required by the GEPSS ontology that is more abstract. For instance, to answer the CQ1 of the GEPSS ontology requirements quite abstract concepts such as infrastructure or the infrastructure name are required. These concepts are included in the *infrastructure ontology module*[6] (placed in the *common-domain layer*), which was also selected for reuse.

Fourth, the selected modules of DABGEO were reused by the ontology engineers to develop the GEPSS ontology. Apart from the *provided energy type* and *infrastructure* ontology modules, the reused modules include: energy usage cost ontology,[7] distributed energy sources state ontology,[8] populated places ontology,[9] energy storage systems ontology,[10] energy generation systems performance ontology,[11] device operation ontology,[12] distributed energy sources ontology[13] and individual user ontology.[14] The reuse of the ontology elements of the aforementioned modules was conducted by referencing such elements or by importing the reused modules as a whole (through the *owl:imports* statement).

In addition, ontology reengineering activities had to be performed over some of the reused modules to adapt the reused knowledge to the GEPSS ontology requirements. New knowledge had to be added and unnecessary knowledge was pruned. Unlike the imported modules, the adaptation of the reengineered modules required making a local version of them apart from the online version. Thus, the reengineered modules lose the connection with the original module in the case it is updated [13]. In case a new version of a DABGEO module is released, the

5 http://www.purl.org/dabgeo/domain-task/application_type/home_energy_assessment_device_control/providedenergytype

6 http://www.purl.org/dabgeo/common-domain/infrastructure
7 http://www.purl.org/dabgeo/domain-task/application_type/home_energy_assessment_device_control/energyusagecost
8 http://www.purl.org/dabgeo/domain-task/smart_grid_scenario/smart_grid_dr/dersstate
9 http://www.purl.org/dabgeo/domain-task/smart_grid_scenario/building_district_city/populatedplaces
10 http://www.purl.org/dabgeo/variant-domain/energystoragesystems
11 http://www.purl.org/dabgeo/variant-domain/generationsystemsperformance
12 http://www.purl.org/dabgeo/common-domain/deviceoperation
13 http://www.purl.org/dabgeo/common-domain/ders
14 http://www.purl.org/dabgeo/variant-domain/individualuser

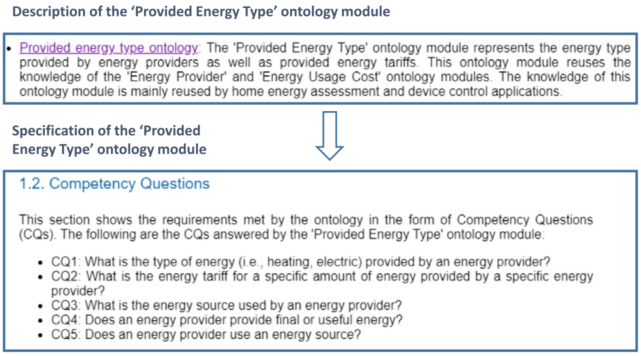

Fig. 4. Analysis of the provided energy type ontology module.

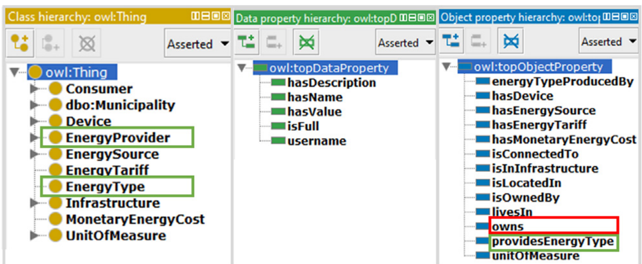

Fig. 5. Class and property structure of the GEPSS ontology.

local modules developed by reengineering that module should be also updated. However, the structure of DABGEO enables to add new knowledge without needing to produce a new version of DABGEO modules. New knowledge can be added as new modules that extend specific modules. The new modules will import the knowledge from already implemented modules, so there will be no need to modify the latter. Thus, the updates in DABGEO are unlikely to affect the reengineered modules [15].

As a result, Fig. 5 shows the class and property structure of the GEPSS ontology in Protégé. It includes the classes and properties from the modules selected for reuse. For instance, the GEPSS ontology includes the ***EnergyProvider*** and ***EnergyType*** classes and the ***providesEnergyType*** property (marked in green in Fig. 5) from the *provided energy type ontology module*. The GEPSS ontology also includes the elements that were added to satisfy the rest of the ontology requirements. For example, the *provided energy type ontology module* does not include the necessary elements to answer the CQ2 of the ontology requirements: *What infrastructures owns an energy provider?*. Hence, the ***owns*** property (marked in red in Fig. 5) was created in the ontology. This property relates the energy providers with the infrastructures they own.

Finally, the GEPSS ontology was tested to check whether it meets the defined requirements. It was instantiated with the data required by the GEPSS and it was loaded into the semantic repository. Then, SPARQL queries were executed against the semantic repository to check whether the obtained results were correct and the ones required by the user interface.

6. DABGEO evaluation

DABGEO was evaluated to determine whether it provides a balance between reusability and usability, since it is the main objective of the ontology.

The balance of reusability–usability of an ontology is demonstrated by showing that it reduces the ontology reuse effort in different applications [14]. In particular, the reusability of an ontology is demonstrated by reusing it in different application types, as was done with well-known reusable ontologies developed in other domains (i.e., [11,29]). The usability is demonstrated by showing that the effort of reusing the ontology to satisfy the requirements of a specific application is reduced [14].

Considering this, the evaluation of DABGEO has focused on determining if it reduces the ontology reuse effort in different energy management applications compared to a global ontology that was not designed to prioritize the balance of reusability–usability.

To evaluate these aspects, we conducted an experiment to measure the reuse effort of DABGEO in two ontology-based energy management applications that operate in different infrastructures. These applications correspond to the GEPSS and EPAS systems introduced at the beginning of Section 5. Two ontology engineers reused separately the ontology modules of DABGEO to develop ontologies that satisfy the knowledge requirements of each application. One of the ontology engineers was part of the ontology development team. Regarding their background, both engineers have knowledge about ontology engineering and have previously contributed in the development of ontologies for specific applications. They have worked in projects related to the energy domain.

The ontology reuse effort of DABGEO was compared with the reuse effort of a global energy ontology which does not prioritize the balance of reusability–usability. Since the OEMA ontology network represents the same energy data domains as DABGEO, the effort of reusing DABGEO in the energy management applications was compared with the effort of doing so with OEMA. In this way, we prevented other factors apart from the ontology design that affect the ontology reuse effort from influencing the result of the experiment (i.e., ontology documentation [30] or represented knowledge). It is worth mentioning that the experiment was limited to develop a specific part of the GEPSS and EPAS ontologies to limit the duration of the experiment.

The following subsections describe the conducted experiment: (1) how the ontology reuse process of both ontologies in each application was performed and how the effort of this process was quantified and (2) the results of the reuse process.

6.1. Ontology reuse process and effort quantification

The ontology reuse process was performed and its effort quantified by taking as reference the ONTOCOM ontology engineering cost model [30,31]. ONTOCOM is applied to estimate the ontology building, reuse and maintenance effort.

6.1.1. Ontology reuse process

Firstly, the ontology engineers were provided with the knowledge requirements of the GEPSS and EPAS ontologies in the form of CQs.

Then, DABGEO and OEMA were reused to develop the GEPSS and EPAS ontologies, which were developed with Protégé. During the ontology reuse process, the ontology engineers could access the documentation and ontology files of DABGEO. It is important to mention that the ontology engineers did not have the chance to develop the GEPSS and EPAS ontologies from scratch or reusing other ontologies, since the experiment was conducted to compare the reuse effort of DABGEO and OEMA. DABGEO and OEMA were reused following the main phases of the ontology reuse process defined by the ONTOCOM model [30]:

1. **Ontology understanding and evaluation:** in this phase, the ontology engineers analysed the description of DABGEO and OEMA modules. Then, the specification of the modules that may met the requirements of the GEPSS and EPAS ontologies was analysed. In addition, the ontology engineers

searched in Protégé for the ontology elements of DABGEO and OEMA required to answer the CQs of the GEPSS and EPAS ontologies. To speed up the search, ontology engineers used the Protégé search engine. They entered the keywords of each CQ to search for the elements that make it possible to answer it. Finally, modules that meet the GEPSS and EPAS ontology requirements were selected for reuse.

2. **Ontology customization:** in this phase, the ontology engineers reused the selected modules to develop the GEPSS and EPAS ontologies. They had the option to reference the elements of the selected modules or to import the selected ontologies as a whole. The ontology engineers also conducted the necessary ontology reengineering activities to adapt the reused knowledge to the requirements of the developed ontologies. In particular, they conducted the following reengineering activities: knowledge addition, class hierarchy restructuring, ontology pruning, property modification and ontology module extraction. For a more detailed definition of each ontology reengineering activity the reader should refer to [13].

Each ontology engineer conducted the ontology reuse phases to develop the ontologies of the GEPSS and EPAS systems twice: reusing OEMA and reusing DABGEO. Once DABGEO or OEMA have been reused in one application, the ontology reuse process of the other ontology in the same application will be simplified because the second time the application ontology is developed the ontology requirements are known. Hence, it should be noted that OEMA and DABGEO were reused in different order in each application to minimize the impact of this aspect in the experiment. In particular, the ontology understanding and evaluation phase was firstly conducted with OEMA in the GEPSS ontology reuse process, while this phase was firstly conducted with DABGEO in the EPAS ontology reuse process. The ontology customization phase was conducted in the opposite order.

Finally, it is worth mentioning that the ontologies developed by the ontology engineers were evaluated to check whether they met the GEPSS and EPAS requirements. The developed ontologies were loaded into the semantic repository of each system. Then, SPARQL queries were executed against the semantic repository to check whether the obtained results were correct and met the requirements of the GEPSS and EPAS systems.

6.1.2. Ontology reuse effort quantification

The effort of performing any ontology activity is the time required to complete the activity [32]. Considering this, the ontology reuse effort was quantified as the time required to perform the each ontology reuse phase.

In addition, we analysed the ontology reengineering activities conducted by the ontology engineers to adapt DABGEO and OEMA to the GEPSS and EPAS ontology requirements. We also analysed the number of ontology elements affected by these reengineering activities. The purpose of this analysis is to show in more detail the effort required to reuse OEMA and DABGEO.

6.2. Ontology reuse results

Table 2 shows the average time needed by the ontology engineers to conduct each ontology reuse phase with DABGEO and OEMA in each energy management application. The ontology reuse phases took less time with DABGEO in both energy management applications. It is worth mentioning that DABGEO reduced the ontology reuse effort with both engineers.

Fig. 6 shows the ontology reengineering activities conducted by the ontology engineers to adapt OEMA and DABGEO to the GEPSS ontology requirements. Fig. 6 also shows the average number of ontology elements affected by the ontology reengineering activities. These elements are ontology modules, (in the case of the ontology module extraction activity), classes, properties and axioms. The reuse of both ontologies required similar activities, which affected to similar number of ontology elements. The case of ontology module extraction and ontology pruning activities was different. Both activities are performed to discard unnecessary knowledge [13] and depending on the size of the reused ontology they may require a significant effort.

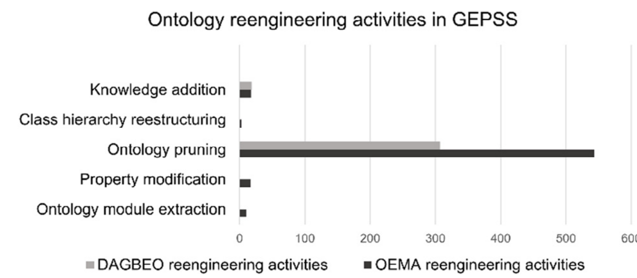

Fig. 6. Ontology reengineering activities in the GEPSS.

In particular, the adaptation of OEMA to the GEPSS requirements required to extract ontology modules from the OEMA ontologies, since they are quite extensive. The extracted modules included the knowledge necessary for the GEPSS ontology. In addition, ontology knowledge was pruned from these modules. The pruned knowledge corresponds to the one not needed to develop the GEPSS ontology. In the case of DABGEO, only the modules that address the GEPSS ontology requirements were reused.

Therefore, no ontology module extraction was required. The modules from DABGEO contain less knowledge than the OEMA ontologies, and the knowledge they include is abstract or specific knowledge reused by certain applications. Hence, less elements were pruned from DABGEO modules. It is worth mentioning that the results were similar in the EPAS case.

7. Discussion

Considering the results of the experiment conducted in Section 6, DABGEO could be adapted to fit the requirements of various energy management applications deployed in different Smart Grid scenarios. Hence, we can state that DABGEO could be reused in different application contexts within the energy domain.

On the other hand, the reuse effort of DABGEO was lower than the reuse effort of OEMA in both applications. This effort reduction was more remarkable in the ontology customization phase. In particular, with respect to OEMA, DABGEO reduced the ontology reengineering activities required to adapt the ontology to the GEPSS and EPAS knowledge requirements. The ontology understanding and evaluation phase required similar effort in the reuse process of DABGEO and OEMA, since DABGEO includes many modules. DABGEO enables to analyse and reuse only the modules related with the requirements of the application ontology. However, ontology developers had to analyse multiple modules to understand and analyse in detail the knowledge that can be reused to develop the ontology of each application.

Once the initial understanding process of DABGEO is completed, ontology engineers can reuse only the necessary knowledge when adding new knowledge to the application ontology to address new requirements. They can also reuse only the necessary knowledge to develop new ontologies. Hence, the effort

Table 2
Average ontology reuse effort.

	Ontology understanding and evaluation phase		Ontology customization phase		Total time	
	GEPSS	EPAS	GEPSS	EPAS	GEPSS	EPAS
OEMA ontology network	1.7 person-hours	1.3 person-hours	2.9 person-hours	2.8 person-hours	**4.6 person-hours**	**4.1 person-hours**
DABGEO ontology	1.6 person-hours	1.4 person-hours	2.6 person-hours	1.4 person-hours	**4.2 person-hours**	**3 person-hours**

reduction will be more remarkable when reusing DABGEO to maintain the ontology or to develop new ones that reuse similar DABGEO modules.

Considering this, DABGEO reduced the ontology reuse effort in the GEPSS and EPAS, and it is likely to keep the reuse effort moderate in new applications. Thus, we can state that DABGEO was usable for specific energy management applications.

In conclusion, the experiment shows that DABGEO could be reused in two energy management applications reducing the ontology reuse effort. Hence, DABGEO provides a better balance between reusability and usability than OEMA, at least for the use cases shown in this paper.

8. Conclusion and future work

In this article we have presented and described DABGEO (current version 1.0), a reusable and usable global ontology for the energy domain. DABGEO can be reused by ontology engineers to develop ontologies for specific energy management applications. It provides a common representation of the energy domains represented heterogeneously by the already available energy ontologies. In contrast to previous global energy ontologies, DABGEO follows reusable and usable ontology design principles to provide a balance of reusability–usability. In particular, it classifies the energy domain knowledge into different abstraction layers that separate the knowledge reused by most applications from the knowledge reused by specific application types. This approach reduces the activities needed to adapt the ontology to different application requirements, thus reducing the ontology reuse effort in different applications.

Two ontology engineers reused separately DABGEO in two energy management applications to demonstrate its balance of reusability–usability. The reuse effort of DABGEO was compared with the effort of reusing another global energy ontology which does not prioritize the balance of reusability–usability: the OEMA ontology network. The results show that DABGEO could be adapted to each application. In addition, with respect to OEMA it reduced the ontology reuse time and adaptation changes, facilitating the ontology reuse process.

As future work, in the DABGEO home page we plan to integrate algorithms that semi-automatically check whether a set of ontologies meet specific requirements [33]. These algorithms would help to reduce the understanding effort that DABGEO requires due to its large number of modules.

On the other hand, the knowledge represented by DABGEO is subject to the domain of energy, as it reuses the knowledge of energy ontologies developed for specific applications. DABGEO could be aligned with these ontologies. In particular, links could be stablished between the equivalent knowledge of DABGEO and existing energy ontologies. These links would enable the knowledge exchange between new applications that use DABGEO vocabularies and legacy applications that use the vocabularies from existing ontologies.

Finally, the structure of DABGEO may be modified when new energy management applications require the knowledge from other energy domains than the ones represented by the ontology. Therefore, a layer that represents domain independent knowledge that can be extended with the knowledge from different domains should be added at the top of the layered structure of DABGEO. In this way, the rest of the layers would extend the domain independent knowledge and the ontology structure would remain consistent when the knowledge of new energy domains is added. Thus, the maintenance of DABGEO would be facilitated [14]. As far as we know, ontology design patterns (modelling solutions to solve recurrent ontology design problems [34]) are applied to represent the domain independent knowledge within layered ontologies [14]. Hence, we will consider the use of ontology design patterns to represent this knowledge in DABGEO.

CRediT authorship contribution statement

Javier Cuenca: Conceptualization, Methodology, Software, Validation, Data curation, Writing - original draft, Writing - review & editing, Visualization. **Felix Larrinaga:** Conceptualization, Investigation, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition. **Edward Curry:** Conceptualization, Resources, Writing - review & editing, Supervision, Project administration.

Acknowledgements

This work has been developed by the Software and Systems Engineering group of MGEP and supported by the Department of Education, Universities and Research of the Basque Government under the projects Ikerketa Taldeak (Grupo de Ingeniería de Software y Sistemas, no. IT1326-19).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] X. Fang, S. Misra, G. Xue, D. Yang, Smart grid - The new and improved power grid: A survey, IEEE Commun. Surv. Tutor. 14 (4) (2012) 944–980.
[2] J. Cuenca, F. Larrinaga, L. Eciolaza, E. Curry, Towards cognitive cities in the energy domain, in: E. Portmann, M. Tabacchi, R. Seising, A. Habenstein (Eds.), Designing Cognitive Cities, in: Studies in Systems, Decision and Control, vol. 176, Springer, 2019, pp. 155–183.
[3] J. Hebeler, M. Fisher, R. Blace, A. Perez-Lopez, Semantic Web Programming, John Wiley & Sons, 2011.
[4] V. Corrado, I. Ballarini, L. Madrazo, G. Nemirovskij, Data structuring for the ontological modelling of urban energy systems: The experience of the SEMANCO project, Sustain. Cities Soc. 14 (1) (2015) 223–235.
[5] M.J. Kofler, C. Reinisch, W. Kastner, A semantic representation of energy-related information in future smart homes, Energy Build. 47 (2012) 169.179.
[6] L. Daniele, M. Solanki, F. Den Hartog, J. Roes, Interoperability for smart appliances in the IoT world, in: P. Groth, et al. (Eds.), The Semantic Web - ISWC 2016. ISWC 2016, in: Lecture Notes in Computer Science, vol. 9982, Springer, Cham, 2016, pp. 21–29.
[7] S. Gillani, F. Laforest, G. Picard, A generic ontology for prosumer-oriented smart grid, in: K. Selçuk Candan, et al. (Eds.), Proceedings of the Workshops of the EDBT/ICDT 2014 Joint Conference (EDBT/ICDT 2014). CEUR Workshop Proceedings, vol. 1133, CEUR-WS, 2014, pp. 134–139.
[8] G. Burel, L.S.G. Piccolo, H. Alani, Energyuse - a collective semantic platform for monitoring and discussing energy consumption, in: P. Groth, et al. (Eds.), The Semantic Web – ISWC 2016. ISWC 2016, in: Lecture Notes in Computer Science, vol. 9982, Springer, Cham, 2016, pp. 257–272.

[9] M. Jarrar, R. Meersman, Ontology engineering - the dogma approach, in: T.S. Dillon, E. Chang, R. Meersman, K. Sycara (Eds.), Advances in Web Semantics I, in: Lecture Notes in Computer Science, vol. 4891, Springer, Berlin, Heidelberg, 2008, pp. 7–34.
[10] N. Choi, I.Y. Song, H. Han, A survey on ontology mapping, SIGMOD Rec. 35 (3) (2006) 34–41.
[11] H. Chen, F. Perich, T. Finin, A. Joshi, SOUPA: Standard ontology for ubiquitous and pervasive applications, in: The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004, IEEE, 2004, pp. 258–267.
[12] E. Paslaru-Bontas, A Contextual Approach to Ontology Reuse: Methodology, Methods and Tools for the Semantic Web, Freie (Ph.D. thesis), Universität Berlin, Germany, 2007.
[13] M.C. Suárez-Figueroa, NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse (Ph.D. thesis), Universidad Politécnica de Madrid, Spain, 2010.
[14] J. Morbach, A. Wiesner, W. Marquardt, OntoCAPE-A (re)usable ontology for computer-aided process engineering, Comput. Chem. Eng. 33 (10) (2009) 1546–1556.
[15] J. Morbach, A. Yang, W. Marquardt, OntoCAPE-A large-scale ontology for chemical process engineering, Eng. Appl. Artif. Intell. 20 (2) (2007) 147–161.
[16] G. Klinker, C. Bhola, G. Dallemagne, D. Marques, J. McDermott, Usable and reusable programming constructs, Knowl. Acquis. 3 (2) (1991) 117–135.
[17] J. Cuenca, F. Larrinaga, E. Curry, Experiences on applying SPL engineering techniques to design a (Re) usable ontology in the energy domain, in: Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, 2019, KSI Research Inc. and Knowledge Systems I, 2019, pp. 606–611.
[18] D. Thakker, V. Dimitrova, L. Lau, R. Denaux, S. Karanasios, F. Yang-Turner, A priori ontology modularisation in ill-defined domains, in: C. Ghidini, et al. (Eds.), Proceedings of the 7th International Conference on Semantic Systems. ACM International Conference Proceeding Series, Association for Computing Machinery, 2011, pp. 167–170.
[19] M. D'aquin, Modularizing ontologies, in: M. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, A. Gangemi (Eds.), Ontology Engineering in a Networked World, Springer, Berlin, Heidelberg, 2012, pp. 213–233.
[20] E. Curry, J. O'Donnell, E. Corry, S. Hasan, M. Keane, S. O'Riain, Linking building data in the cloud: Integrating cross-domain building data using linked data, Adv. Eng. Inform. 27 (2) (2013) 206–219.
[21] T.G. Stavropoulos, D. Vrakas, D. Vlachava, N. Bassiliades, BOnSAI: A smart building ontology for ambient intelligence, in: WIMS '12: Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics. ACM International Conference Proceeding Series, Association for Computing Machinery, 2012, pp. 1–12.
[22] E. Blomqvist, P. Thollander, An integrated dataset of energy efficiency measures published as linked open data, Energy Effic. 8 (6) (2015) 1125–1147.
[23] J. Cuenca, F. Larrinaga, E. Curry, A unified semantic ontology for energy management applications, in: L. Bozzato, et al. (Eds.), Joint Proceedings of the Web Stream Processing workshop, WSP 2017, and the 2nd International Workshop on Ontology Modularity, Contextuality, and Evolution, WOMoCoE 2017, co-located with 16th International Semantic Web Conference, vol. 1936, CEUR-WS, 2017, pp. 86–97.
[24] M. Lefrançois, Planned ETSI SAREF extensions based on the W3C & OGC SOSA/SSN-compatible SEAS ontology patterns, in: A. Fensel, L. Daniele (Eds.), Joint Proceedings of SEMANTiCS 2017 Workshops co-located with the 13th International Conference on Semantic Systems, SEMANTiCS 2017. CEUR Workshop Proceedings, vol. 2063, 2017.
[25] W3C OWL Working Group, Owl 2 web ontology language document overview, OWL 2 Web Ontol. Lang. no. December, 2012, pp. 1–7.
[26] C.J.H. Mann, The Description Logic Handbook – Theory, Implementation and Applications, vol. 32, no. 9/10, Emerald Group Publishing Limited, 2003.
[27] H. Zhang, Y.F. Li, H.B.K. Tan, Measuring design complexity of semantic web ontologies, J. Syst. Softw. 83 (5) (2010) 803–814.
[28] M. Grüninger, M.S. Fox, M. Gruninger, Methodology for the design and evaluation of ontologies, in: International Joint Conference on Artificial Inteligence, IJCAI95, Workshop on Basic Ontological Issues in Knowledge Sharing, 1995, pp. 1–10.
[29] C. Schlenoff, et al., An ieee standard ontology for robotics and automation, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2012, pp. 1337–1342.
[30] E.P.B. Simperl, C. Tempich, Y. Sure, Ontocom: a cost estimation model for ontology engineering, in: I. Cruz, et al. (Eds.), The Semantic Web - ISWC 2006. ISWC 2006, in: Lecture Notes in Computer Science, vol. 4273, Springer, Berlin, Heidelberg, 2006, pp. 625–639.
[31] E. Simperl, T. Bürger, S. Hangl, S. Wörgl, I. Popov, ONTOCOM: A reliable cost estimation method for ontology development projects, J. Web Semant. 16 (2012) 1–16.
[32] E.P. Bontas, M. Mochol, R. Tolksdorf, Case studies on ontology reuse, in: Proceedings of the IKNOW05 International Conference on Knowledge Management, vol. 74, 2005, pp. 345–353.
[33] A. Fernández-Izquierdo, R. García-Castro, Themis: A tool for validating ontologies through requirements, in: Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, 2019, KSI Research Inc. and Knowledge Systems I, 2019, pp. 573–578.
[34] A. Gangemi, V. Presutti, Ontology design patterns, in: S. Staab, R. Studer (Eds.), Handbook on Ontologies. International Handbooks on Information Systems, Springer, Berlin, Heidelberg, 2009, pp. 221–243.