

RESEARCH ARTICLE

Semi-Automatic Ontology Development Framework for Building Energy Data Management

ZHIYU PAN¹, (Member, IEEE), YUTING GAO¹, FERDINANDA PONCI¹, (Senior Member, IEEE), AND ANTONELLO MONTI^{1,2}, (Senior Member, IEEE)

¹Institute for Automation of Complex Power Systems, E.ON Energy Research Center, RWTH Aachen University, 52074 Aachen, Germany

²Fraunhofer FIT, 53757 Sankt Augustin, Germany

Corresponding author: Zhiyu Pan (zhiyu.pan@eonerc.rwth-aachen.de)

This work was supported in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Grant 501865131; in part by the German Federal Government, in part by the German State Government; and in part by the Joint Science Conference (GWK) for support as part of the National Research Data Infrastructure for the Interdisciplinary Energy System Research (NFDI4Energy) Consortium.

ABSTRACT With the ongoing digital transformation and multi-domain interaction occurring in the buildings, a huge amount of heterogeneous data is generated and stored on a daily basis. To take advantage of the gathered data and help better decision makings, suitable methods are needed to meet the demand for building operations and reinvestment planning. Ontology, which provides not only the vocabulary of a certain domain but also the relationship between each other has been used in multiple engineering fields to manage heterogeneous data. A plethora of ontology development methodologies have been developed in the last decade, whereas those methods are still really time-consuming and in a low degree of automation. In this paper, we approach the problem by first presenting a semi-automatic ontology development framework that integrates existing automatic ontology tools and reuses existing ontology and data model. Based on this framework, we create a building energy management ontology and evaluate the data coverage of several real-life data sets.

INDEX TERMS Building energy management, data management, data model, ontology.

I. INTRODUCTION

With the development of information and communication technologies, a huge amount of disparate information is available, thus managing heterogeneity among various information resources has been challenging since then. It is acknowledged in most data management research that the thorny issue of semantic heterogeneity, which comprises handling variations in meaning or ambiguities in entity interpretation, remains a challenge [1].

For example, the energy domain is a broad field that involves a series of definitions such as energy sources, supply, and consumption. The energy consumption in buildings is influenced by many factors, including building structure, environmental conditions, and the operation of electronic

components like lighting and HVAC systems. Therefore, when analyzing the data related to energy consumption in a building, a substantial variety of information necessitates processing. To make the best use of the information at hand, it is important and necessary to handle data information more comprehensively and systematically. The utilization of multi-domain ontology facilitates the establishment of a shared representation for data across disparate domains, thereby enabling facile comprehension and processing of said data. One of the goals of this paper is to address the heterogeneity problem for building energy data management by developing a cross-domain ontology.

As proposed in [2], ontologies that generate explicit specifications of conceptualizations have become widely adopted across the engineering community. The use of ontologies, as a component of knowledge-based systems, has been widely used for the effective management of knowledge

The associate editor coordinating the review of this manuscript and approving it for publication was N. Prabaharan¹.

in the domain of discourse within organizations [3]. It formally describes knowledge as a set of concepts within a domain, as well as the relationships that hold between them. As a result, ontologies introduce a reusable and shareable knowledge representation as well as the potential to incorporate additional domain-specific knowledge. Thus, its usage guarantees a consensus over information and makes explicit domain assumptions, which allows organizations to manage their data better.

A. PROBLEM STATEMENT

Current existing ontologies focus on one specific application. Unfortunately, there is no uniform ontology that can be used in all cross-domain applications for building energy management [4]. By reusing one ontology, it is hard to ensure the coverage of terminology and also causes ambiguities. In most cases, multiple ontologies are required for one application. Additionally, manually merging ontologies from different domains is time-consuming and requires domain knowledge. Therefore, an ontology development framework that simplifies the ontology reusing process is required. We have compared the ontology development methodologies mentioned in [5], [6], [7], [8], [9], [10], [11], and [12]. These methods provide a better understanding of the process of ontology development and highlight the importance of ontology reuse and ontology merging. However, the precise methodologies and tools for identifying appropriate existing ontologies for specific use cases and facilitating their reuse in the ontology development process remain undetermined. Additionally, the reuse of other knowledge sources is proposed in [10]. However, this paper does not demonstrate the process of how to transform other knowledge sources into ontologies. In summary, the following problems should be covered in our method implementation:

- 1) How to find useful ontologies to reuse?
- 2) How to reuse other knowledge sources (e.g. data model) to develop an ontology?
- 3) What kind of tools and methods are available to automate ontology development?

B. CONTRIBUTION

The first contribution of this paper is the introduction of a novel semi-automatic ontology development framework. We focus on the reuse of existing knowledge sources (e.g. ontology and data model) automatically. Furthermore, we analyze the existing automatic tools and methods for ontology reuse, matching, and merging, select the possible candidates for ontology development and integrate these tools and methods in the developed framework to automate the ontology development processes, thereby saving the expected effort for the development of ontologies. For the requirement definition, ontology evaluation and ontology maintenance, human interaction with the framework is still required. Therefore, we define our framework as semi-automatic. Finally, we demonstrate the ontology development framework for

the building energy domain and provide an exemplary case using this ontology development framework to generate the Building Energy Ontology based on SAREF4ENER [13], SAREF4BLDG [14], SBEO [15], SARGON [16], EM-KPI Ontology [17], FIWARE [18], Schema.org [19], and EPC4EU [20].

C. STRUCTURE

In this section, we present the overview of our research, including the motivation, problem statement and contributions. The remainder of the paper is organized as follows: in section II we present the related work in ontology development and introduce preliminary information. In section III, we introduce the ontology development framework. In section IV, we discuss the implementation of developing an ontology in multi-domain. Subsequently, in subsection IV-H, we evaluate the results of our work. Finally, in section V, we conclude our work and suggest possible future work.

II. RELATED WORK

The associated work is divided into two sections: existing methodologies for ontology development and the comparison of data model and ontology.

A. ONTOLOGY DEVELOPMENT METHODOLOGIES

Over the last decade, several researchers have proposed different ontology development methods.

Skeletal Methodology [5], also known as the Enterprise Method, developed by Mike Uschold and Michael Gruninger, is used primarily to build enterprise ontologies. This is the first general methodology for ontology development and provides a rough structure for ontology development, which include identifying purpose, building the ontology, evaluation and documentation. The method in [6] further developed the Skeletal Methodology, which is structured in three phases: Management phase, Development phase and Maintenance phase. In the development phase, ontology reuse is considered in the conceptualization step. But this is done by the domain expert manually. Additionally, how to find the existing ontology is not described.

The method in [7] is mainly used for domain ontology development. This method is based on iterative design, which allows developers to build an ontology using the tool Protege. Compared with the previous method, this research provides not only the method but also how to use the tool Protege to simplify ontology development.

In [9], the extraction of domain ontologies from text is discussed. This process involves five steps: data source selection, concept learning, domain focus, relationship learning and evaluation. The method adopts a ring structure development, which clarifies the importance of ontology iteration. Moreover, the combination of ontology evaluation and ontology evolution promotes ontology refinement.

The NeOn methodology framework [10] is a scenario-based methodology that defines a set of nine scenarios for building ontologies and ontology networks. Furthermore,

the construction of ontologies and ontology networks is supported by reusing ontologies, non-ontological resources, and ontology design patterns.

In [12], the authors present a framework for the building of multi-aspect ontologies that combines the aspect integration approach at various levels. Based on human-machine collective intelligence, the methodology is used to create a multi-aspect ontology for decision support.

To conclude, the mentioned methodologies for ontology development are compared regarding the following characteristics: knowledge search, data model transformation and automatic ontology reuse. In terms of knowledge search, Methontology, NeON, AMOD and Multi-aspect ontology involve the use of existing ontologies. However, those methodologies focus only on the guideline of how to leverage existing ontologies and do not describe how to search for the existing ontology. Only NeOn considered reusing non-ontological resources (e.g. data model). However, no explicit applications are described on how to reuse and how to transform data model into ontology. None of the existing methodologies include automatic ontology reuse.

B. COMPARISON OF DATA MODEL AND ONTOLOGY

Ontology: is regarded as a formal, explicit specification of a shared conceptualization [21]. A conceptual description of the entities, attributes, and connections inside a domain was described as an ontology [3].

Data model: As mentioned in [22], the foundation of a data model is data relationships, data semantics, data constraints and data itself. The specifics of the information to be stored are provided by a data model.

Although both ontologies and data models are partial accounts of conceptualizations [23], they are still distinct from each other. The fundamental focus of an ontology is to specify and share meaning, thus it should be as generic and task-independent as possible [24]. In contrast, the data model is used to describe data and usually gets updated to meet particular new functional requirements, thus it is more task-specific and implementation-oriented. The most commonly used language for data model is the Unified Modeling Language (UML) and for ontology is Ontology Web Language (OWL), Resource Description Framework (RDF) and Resource Description Framework Schema (RDFS) [25]. According to [26], UML is a general-purpose visual modeling language. OWL, which was first proposed in 2004 and later released as OWL2 in 2009, is designed for programs that process information instead of being human-readable [27]. Although UML and OWL have different perspectives, there are significant overlaps and commonalities between them, particularly in the representation of structure (class diagrams). Classes, relationships, properties, packages, types, generalizations, and instances are a few of the components shared between OWL and UML [28]. Both OWL and UML are modeling languages. OWL is a notation for knowledge representation, whereas UML is a notation

for modeling the products of object-oriented software [29]. By transforming the data model, it is possible to reuse the structure and avoid having to spend time building the ontology from scratch.

III. SEMI-AUTOMATIC ONTOLOGY DEVELOPMENT FRAMEWORK

This section presents a semi-automatic ontology development framework, which provides a clear process to guide ontology development based on the existing automatic tools. The following subsections describe the ten steps of our ontology development and the involved methods and tools in each step.

A. ONTOLOGY DEVELOPMENT WORKFLOW

We briefly introduce the ontology development framework, depicted in Figure 1. It can be applied to both the development of single-domain and multi-domain ontologies.

1. *Domain and scope determination* aim to set the stage for developing or enhancing ontologies and determining the scope of the third step Knowledge Search.

2. *Requirements* define the prerequisites that the ontology must satisfy, such as the domain to which it applies and the features that it must include.

3. *Knowledge Search* aims to find the existing knowledge sources in the corresponding domains according to the requirements defined in the previous step. The knowledge sources in the paper are ontology and data model.

4. *Ontology Reuse* select the most suitable ontology from the existing ontologies, which are the output from the Knowledge Search step.

5. *Matching* finds the relationship and overlap of entities between the re-used ontologies and generates the matching results. These results are used as input for the following ontology merging. When there is only one suitable ontology to be reused, this step can be omitted.

6. *Data Model Transformation* means the existing data model in UML is transformed into an ontology language if no suitable ontology can be found for reuse.

7. *Conceptualization* creates the ontology according to the requirements, in case there is no ontology and data model that is suitable for reuse.

8. *Merging* is the process of combining two or more ontologies into a new ontology based on the outputs from the matching step.

9. *Evaluation* includes checking the feasibility of this methodology and whether the created ontology meets the requirements proposed at the beginning.

10. *Maintenance* is to modify the ontology based on the results of the evaluation. If an additional requirement is identified, the process reverts back to the initial step and the iterative loop recommences.

B. REQUIREMENTS

As defined in [30], the ontology requirements can be divided into the following two types: non-functional ontology requirements and functional ontology requirements. The

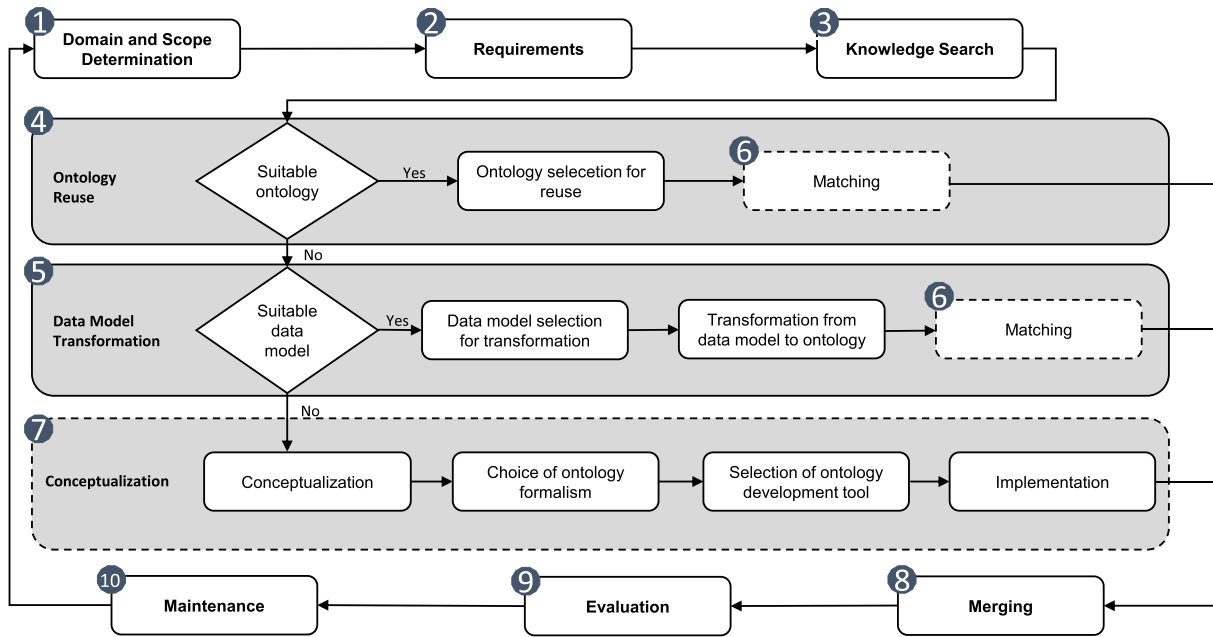


FIGURE 1. Ontology development workflow.

defined requirement is presented as Competency Questions (CQ). CQ are a set of questions that are formulated to capture the domain knowledge of the target application and its stakeholders. They are designed to elicit the requirements and expectations for the ontology and to provide a basis for evaluating whether the ontology meets its intended purpose.

C. KNOWLEDGE SEARCH

Knowledge search includes ontology search and data model search. Using keywords is the common way to search for existing ontology, which is related to the scope and domain. To identify the associated keywords for an ontology, we extract relevant terms (concepts and relationships) that need to model the domain and scope for ontology and fulfill the requirements. A few keyword research tools, like Google Trends [31], Keyword Shitter [32] and AdWord & SEO Keyword Permutation Generator [33], can also be utilized to gain information. The popularity of terms that is searched over time can be shown by using Google Trends. Through the word's variations and similar search phrases, we can further filter down the list of words that are linked. To obtain additional keywords with Keyword Shitter, users only need to enter one or more seed keywords.

An ontology is a formal representation of a set of concepts and the relationships between them. It provides a shared vocabulary for a particular domain. A knowledge graph, on the other hand, is a network of entities and their relationships, represented in a structured way. It is a type of ontology that emphasizes the connections between entities [34]. Because several large knowledge graphs (KGs) have been created in recent years, including Freebase [35], DBpedia [36], OpenCyc [37], Wikidata [38], and YAGO [37],

which provide a significant amount concepts and relationships. To create an ontology using a knowledge graph start by defining the domain of interest and identifying the relevant entities and relationships. This information can then be used to create a formal ontology that captures the key concepts and relationships within that domain.

A significant amount of organized world information is included in large-scale world knowledge graphs like those found on Freebase [35] and DBpedia [36]. Linked Open Vocabularies (LOV) is a gateway to reusable semantic vocabularies on the Web [39]. Ontologies of related domains are also available through LOV.

A part of the ontology search, as mentioned before in section II, data models are another knowledge resource, which provides data concepts and relationships. FIWARE smart data model was developed by a joint collaboration initiative to support common information exchange in cross-sector applications. A smart data model in FIWARE [18] consists of three components: a schema, or technical representation of the model that specifies the technical data types and structure; a written document definition for human readers; and examples of payloads for NGSiv2 and NGSi-LD versions. FIWARE provides information about various domains, of which smart energy, smart environment, and smart sensors are related to our work. Schema.org began with 297 classes and 187 relations and has since expanded to 638 classes and 965 relations [19]. Industry Foundation Classes is another data model related to the building domain and used in architecture, engineering, construction, and facility management industries for the exchange and sharing of information throughout the building lifecycle [40].

D. ONTOLOGY REUSE

Ontology reuse is crucial for ontology development and is recommended in current methodologies and guidelines as a key factor in developing cost-effective and high-quality ontologies [41]. The current problem for Ontology Reuse is multiple ontologies of one domain can be found. Moreover, those ontologies may have some common classes or properties. As shown in [16], different ontologies in energy domains were compared manually in terms of the coverage of energy domain applications. Based on those comparisons, the relevant ontologies are reused.

In our framework, we use an automatic approach to select suitable ontologies among those identified in the Knowledge Search step. The approach in [42] is applied in our framework. The advantage of this approach compared with other approaches [10], [41] is the semantic similarity-based algorithm, which selects suitable ontologies automatically. The first step is to generate a lexical chain with the relevant terms, which are already defined in the Knowledge Search step as keywords. Then, this approach calculates a Global Grade (GG) for each selected ontology after the Knowledge Search step associated with the lexical chain. The GG [43] is used as an indicator to select the suitable ontology and is given by the sum of the Syntactic-Semantic Grade (SSG) and the Semantic Grade (SG). The SSG can be measured by the sum of its relevant word weights (terms centralities).

The Semantic Grade (SG) [44] is calculated by combining the path length (l) between pairs of terms with the depth (d) of their subsumer. Furthermore, the strength of each relationship between the linguistic properties is expressed by a weight assigned to each arc between the nodes in the ontology.

E. DATA MODEL TRANSFORMATION

The Data Model Transformation step is defined in this paper as transforming a data model to an ontology such as entities, attributes and relationships, to corresponding concepts, properties and relationships in an ontology. This transformation involves converting the data model's structural and semantic information into a formal language representation that adheres to the rules and constraints of the ontology's vocabulary and syntax.

In [45], the authors present Chowlk which is a converter to transform digital UML-based ontology diagrams into OWL. Chowlk web application transforms an ontology conceptualization from diagrams.net into an OWL implementation. The converter identifies concepts, object properties, datatype properties, and restrictions between those elements. As soon as the corresponding associations are detected and created, Chowlk applies the OWL language to write the implementation. Finally, the generated ontology is provided in two downloadable formats: Turtle and RDF/XML. Therefore, Chowlk is used to transform the data model into an ontology in our framework.

F. MATCHING

The outputs from the Ontology Reuse and Data Model Transformation steps, which are ontologies related to the domain and scope of the ontology development, are employed as inputs for the Matching step. The goal of the Matching step is to find the correlation between different ontologies and solve the conflict in concepts, properties or axioms. Semantic heterogeneity can be solved through ontology matching. The focus of many matching systems focuses on combining and extending the known methods. There are a number of popular matching algorithms, such as edit distance, WordNet matchers and iterative similarity matchers [46], [47]. It is important to note that there are different methods for ontology matching and ontology merging, and here only one of the most suitable methods for implementation is discussed.

Since the edit distance algorithm relies on the structure of the text itself, semantic-level information cannot be matched. The semantics of the concepts and relationships in an ontology can be defined by its reference to *WordNet* [48]. However, the solution is not ideal since WordNet lacks some specific words. Taking the phrase *energy consumption* as an example, it is possible to find individual words energy and consumption, but unable to find the combination of them. Moreover, WordNet is not able to directly attribute a connection between energy consumption and energy use. In WordNet, a synset consists of a sense, a lexeme, and a number. A synset is a definition of a word's meaning. In real life, a single word can be used in different scenarios, for this reason cross-linguistic recognition and sentence meaning analysis are focused on determining which sense corresponds to a given word in a given context since this effect is important to real-world understanding.

Therefore, COMA is used in our framework to match the different ontologies. COMA is a one-to-one matching system for ontology matching that was developed over the last decade. It has several characteristics as follows [49], [50]:

- **Configuration Engine** supports both a manual configuration and an automatic configuration.
- **Enrichment Engine** performs an enrichment based on the mapping result and involves iterating through each correspondence in the mapping and passing it to each strategy.
- **User Interface** is straightforward and easy to use. It can be obtained and installed directly through GitHub.

Compared with other methods, COMA provides an open-source and visual tool for ontology matching. Furthermore, COMA can be set up manually to change the matching mechanism in order to achieve exact matches. Therefore, we choose COMA as the tool for ontology matching in this step.

G. CONCEPTUALIZATION

If there are still concepts and relationships, which cannot be found in the existing ontology and data models, we have to develop those in the Conceptualization step. Because the

conceptualization methods have been already well-developed in previous studies and are not the focus of our framework. Therefore, the authors suggest applying the methodology in [51] which provides the comparison of different methods and [7] which provides the guideline and example of using Protege [52], to define the new conceptualization.

H. MERGING

A general definition of ontology merging is the process of combining two or more ontologies into a single ontology [53]. Based on [49], [54], [55], we analyze the existing tools of ontology merge to identify the possible candidate to integrate into our framework. Although in literature various methods of ontology merging are presented, there still are relatively few direct tools based on those methods available that can be easily reused. In total, three ontology merging tools are available at the time of writing, which are Protege [52], PROMPT [56] and COMA [49].

Protégé is a widely used open-source tool for developing and managing ontologies. It provides a user-friendly interface for building ontologies using a variety of ontology languages, such as OWL and RDF. Protégé is highly extensible and allows users to customize the tool by developing plugins that add new functionality.

The PROMPT tool has been implemented as an extension to the Protege ontology editing environment. In the PROMPT suite, there are tools for many of the tasks that are required for managing multiple ontologies: iPrompt is an interactive ontology merging tool, AnchorPrompt uses non-local context for semantic matching, PromptFactor is a tool for factoring out semantically independent sub-ontologies, PromptDiff is used for versioning of ontologies, and updating ontology libraries [56].

A major addition to COMA is the inclusion of an ontology merging component that consumes the match mapping as input and produces an integrated ontology as output, called merged ontology. The main technique of it is called Automatic Target-Driven Ontology Merging (ATOM) [57].

After analyzing the merging tools, the function of ontology merging in COMA is not publicly available. In addition to this, PROMPT is only supported by Protege 3.x and previous versions. Therefore, Protege is selected as the ontology merging tool in our framework. We merged two of the ontologies with the support of Protege in this step and adjusted them based on the matching results from COMA. And the merged ontology is saved as an OWL file directly with Protege.

I. EVALUATION

Several ontology evaluation tools [58], [59], [60], [61] have been developed previously. OOPS is a tool designed to check problems with ontologies that may cause modeling errors [61]. The “Pitfall Scanner” module in OOPS checks the pitfalls in the ontology according to the pre-defined catalog. Ontology OWL code provides information about 32 pitfalls

that can be detected automatically. The “Suggestion Scanner” module generates some modeling suggestions during the scanning phase, in order to identify ontology elements that are at risk of errors. In our framework, OOPS is used to automatically check the pitfall of the merged ontology from the previous step.

Apart from the automatic ontology evaluation tool, Ontologies can be evaluated in different ways. The competency questions, which are defined in the Requirements step, can be checked by the domain expert if the developed ontology answers the questions. One common indicator is coverage, which describes how well the data is covered by the developed ontology. Therefore, the coverage test is also executed after OOPS in the Evaluation steps.

J. MAINTENANCE

Ontology maintenance refers to the ongoing process of managing and updating an ontology to ensure that it remains accurate, relevant and useful over time. In our proposed method, the ontology maintenance process consists of three parts, i.e., adjusting the ontology based on the evaluation results, ontology publication, and further maintenance after the publication.

After adjusting the ontology in response to the issues identified in the ontology evaluation, the ontology should be released online and accessible to the public. Moreover, the developed ontology is published with both human-readable and machine-readable documentation. To create machine-readable formats of information, the standard ontology language OWL is used during the implementation process. To generate the ontology with standard ontology languages, auxiliary tools such as Protege [52] can be used. In addition, to generate a human-readable document, tools such as Widoco [62], OnToology [63], generate HTML documents from OWL and RDF ontology files. OnToology is a web-based tool designed to automate the ontology maintenance process in collaborative environments. OOPS, Widoco, AR2DTool, and GitHub are all integrated into OnToology. A diagram, HTML documentation, and an evaluation report are produced by OnToology using the first three systems. With GitHub, the repository is cloned, OnToology users are added as collaborators, webhooks are created (that notify OnToology of repository changes), and the outputs from the integrated systems are aggregated into pull requests to the repository, where the maintainer can review and merge.

After passing all previous steps in Figure 1, the developed ontology can be further maintained by adopting changes based on new information. Further maintenance involves tasks such as identifying and correcting errors, adding new concepts or relationships, removing outdated information, and aligning the ontology with changes in the domain. The developed ontology can be adapted to accommodate new requirements and domains and this is also the pre-season why the close loop structure is used in the framework.

IV. IMPLEMENTATION

In this section, the ontology development framework is applied to develop a building energy domain ontology. The goal of the application is to develop a holistic, state-of-the-art AI-powered framework for building energy management and a semantic and business interoperability framework for cross-domain analytics applications. It envisions to become the greatest energy marketplace of big data and services in the building sector and involves 11 pilots from different countries and covers the whole lifecycle of buildings.

A. DOMAIN AND SCOPE DETERMINATION

Developing an ontology is primarily motivated by scenarios related to the application that will make use of it. The purpose of this task is to determine the intended Domain and Scope of the ontology. The application aims to provide a state-of-the-art framework for building energy management. The required ontology should encompass multiple domains, including Energy, Buildings, Device, Weather, Questionnaire, Energy Performance Certificates (EPC), EV Charging Station, Calendar and People. The scope of the developed ontology is to act as an overlay to expose the building energy data structure in a way that makes it easier for users to present and analyze data.

B. REQUIREMENTS

In the following, we provide requirement examples and avoid being exhaustive. The non-functional requirements are:

- 1) The developed ontology must be presented in a common ontology language,
- 2) The developed ontology must be based on the existing knowledge sources.

The functional requirements are described with CQ as follows:

- CQ1. What kind of sensor is used?
 - Temperature, Smart lector, Humidity, Heat, CO₂, CO, Electrical, PM_{1,2.5,10}, Thermal, Domestic Hot Water, Electrical, Water, Gas meter, Flow counter and Frequency sensor.
- CQ2. What is EPC?
 - It is Energy Performance Certificates.

C. KNOWLEDGE SEARCH

In this step, separate searches were conducted on different web pages based on the keywords of the relevant fields. The keywords based on the domain for the developed ontology are {*Energy, Buildings, Device, weather, Questionnaire, EPC, EV Charging Station, Calendar, People*}. In particular, we first search the existing ontology by entering the domain keyword, e.g., energy, into the ontology library like [36], [39] and determine if the retrieved ontology is available.

In Table 1, we summarize the ontologies identified. Based on the tables above, there are some ontologies that cover multiple domains at the same time. The results of the ontology search still showed us ontologies that cover more than one

domain even if we performed a single-domain search in the ontology search. Ontologies with multiple target domains are prioritized, as there can already have connections between the domains within the multi-domain ontologies. After analyzing the result in Table 1, we found that ontologies related to EV Charging Station, Questionnaire, and EPC domain are still lacking. Therefore, we search existing data models for those domains. In FIWARE smart data model [18], the concept and relationship of the EV Charging Station are available. The Questionnaire is available in Schema.org. In [20], the developed data model (EPC4EU) for the EPC of buildings in Europe is given, which is selected to be used as input in Data Model Transformation.

D. ONTOLOGY REUSE

After conducting our ontology search, we have found that most of the searched ontologies (in Table 1) focus on Energy, Buildings and Device domain. To automate the Ontology Reuse step and to select the most suitable ontologies for Energy, Building and Device domain, we use the method in [42]. We manually created the lexical chain with the relevant terms related to Energy, Buildings and Device domain, which is {*Building, Building Operation, Room, Device, Photovoltaic Device, Energy, Smart Metering Observation, Photovoltaic Measurement, Battery, Storage Battery, Device, Storage Battery Measurement, Energy Consumption*}. Based on the lexical chain, we calculated the Global Grad (see subsection III-D) of the identified ontology in Table 1. The results are shown in Figure 2. The top five ontologies with high Global Grad values are selected, i.e., SAREF4BLDG, SAREF4ENER, SARGON, SBEO, and EM-KPI-ontology. Those ontologies contain a comprehensive range of building energy systems, including heating, cooling, lighting, energy storage, and renewable energy systems.

E. DATA MODEL TRANSFORMATION

As mentioned in the previous section, we still lack the ontologies for Questionnaires, EV Charging Stations and EPC domains. To solve this problem, we are in the Data Model Transformation step. FIWARE smart data model provides the concept and relationship of EV Charging Station. Schema.org provides the concept and relationship of Questionnaire. In this step, we used the tool named Chowlk [45] to automatically transfer the data model into an ontology.

An example of the conversion of the EPC4EU data model into an ontology is depicted in Figure 3. EP4EU contains five concepts: Building, Certificate, Certifier, Energy System, and Energy Conversion System. Since we have already determined several ontologies for the building domain in the Ontology Reuse step, we keep only the four concepts: Certificate, Certifier, Energy System, and Energy Conversion System in the UML diagram of EPC4EU. As shown in Figure 3, we use arrows to indicate the corresponding transformations:

TABLE 1. Result of ontology search.

Ontology	Domain	Public Available
SAREF4BLDG [14]	Energy, Building, Device	Yes
SARGON [16]	Energy, Building, Device	Yes
BONSAI [64]	Energy, Building	Yes
Think Home Ontology [65]	Energy, Building, Weather, Person	Yes
Building Topology Ontology [66]	Building	Yes
BCOM [67]	Building, Person	Yes
IFC [68]	Building	No
SBEO [69]	Building, Device, Person	Yes
BIMERR Ontology [70]	Building, Device, Weather	Yes
SAREF4ENER [71]	Energy, Device	Yes
MIRABEL [72]	Energy, Device	No
SEMANCO [73]	Energy, Building	No
ISO 37120 [74]	Energy	Yes
EM-KPI-ontology [17]	Energy, Building, Device, Weather	Yes
FSO [75]	Energy, Device	Yes
SSN [17]	Device	Yes
SOSA [76]	Device	Yes
vCard [77]	Person	Yes
Time Ontology [78]	Calendar	Yes
DBpedia [36]	General	Yes

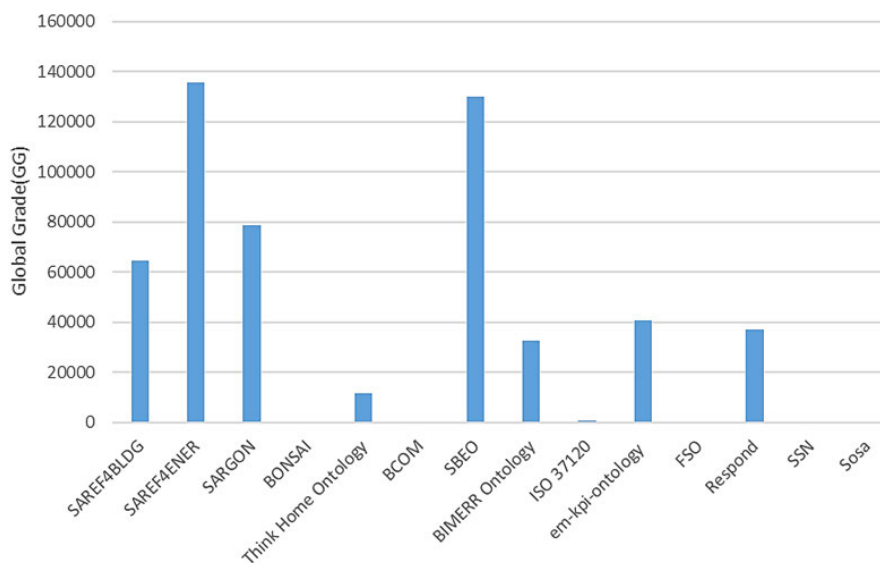


FIGURE 2. Global grade of related ontologies.

- The yellow arrow points that the class in the UML diagram corresponds to the concept class in ontology.
- The blue arrow shows that the class-to-class relationship in the UML diagram is converted into ObjectProperty in the ontology.
- The green arrow indicates that the attributes in the UML diagram are converted into DataProperty in the ontology.

F. MATCHING AND MERGING

We import the top five ontologies selected from the Ontology Reuse step into COMA, and the successfully imported ontology was displayed in the interface Repository under the schema. In the next step, from the imported ontologies, we selected one as the target ontology and one as the resource ontology and finally selected the matching mechanism from the matching. COMA only supports two ontologies as input

to match at the same time. Therefore, five ontologies are pairwise matched.

Based on the matching result, the overlapped concepts and relationships are:

- *power source* in SAREF4BLDG and SAREF4ENER;
- *is measured in* in SAREF and SARGON;
- *relates to property* in SAREF and SARGON;
- *primary current* in SAREF4ENER and SARGON;
- *primary voltage* in SAREF4ENER and SARGON;
- *secondary current* in SAREF4ENER and SARGON;
- *secondary voltage* in SAREF4ENER and SARGON;
- *has function* in SAREF and SARGON;
- *has timestamp* in SAREF and SARGON;
- *has value* in SAREF and SARGON;
- *Location* in SARGON and EM-KPI-ontology.

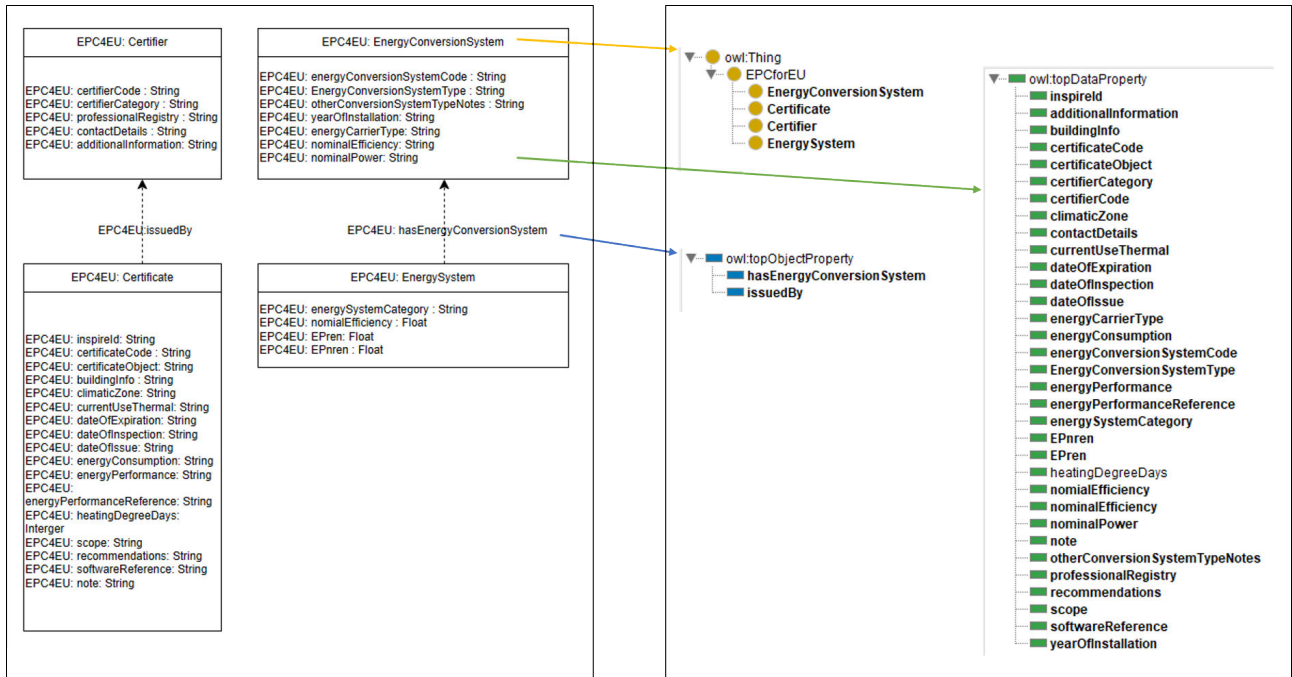


FIGURE 3. Data model of EPC4EU (left) into EPC4EU ontology (right).

The next step is Ontology Merging. As mentioned in previous sections, We use Protege to automatically merge two ontologies and ignore all the overlapping. Afterward, the overlapped concept and relationships based on the matching results are removed with the support of protege. Once all five ontologies are merged, the merged ontology is saved as an OWL file.

To demonstrate the efficacy of the proposed framework, a comparative analysis was conducted between the proposed framework and manual ontology matching and merging techniques. The manual ontology matching process involved the examination of all concepts and relationships within five ontologies, resulting in a total of 9270 elements, in order to identify overlapping terms. Subsequently, the overlapping terms were eliminated from each ontology. For instance, both the SARGON and EM-KPI-ontology contained the concept *Location*. To eliminate this redundant concept, various adjustments were made, including the modification of the IRI (Internationalized Resource Identifier) of *Location*, the alteration of the label assigned to *Location*, the adjustment of relationships associated with *Location*, and the refinement of superclass and subclass definitions related to *Location*. These manual steps can be automated through the utilization of the developed framework.

G. EVALUATION AND MAINTENANCE

OnToology is used to automate the Evaluation and Maintenance step. Table 2 presents a list of pitfalls that have been found, including important and minor errors. As can be seen from the table, the problem lies mainly in the missing domain

TABLE 2. Results of the created ontology evaluation based on OOPS!

Description	Cases	Importance
missing annotations	59	Minor
missing domain or range	78	Important
equivalent properties not declared	5	Important
inverse relationships not declared	67	Minor
different naming conversations	2	Minor
using recursive definitions	7	Important
equivalent classes not explicitly declared	1	Important

or range, which restricts the subject or the object of certain properties. This represents a design decision in ontology implementation. The ontology that has been developed is intended to offer a broad and task-independent conceptualization, intentionally avoiding excessively detailed constraints. Consequently, our framework chooses to ignore missing domain or range pitfalls. The remaining pitfalls are addressed in the final version.

In addition, to test whether the developed building energy ontology fulfills the scope and requirements of building energy management, we utilized the developed ontology to model real-life datasets. The coverage of ontology is chosen as the indicator. Table 3 shows the coverage of the building energy domain ontology, which contains 11 large-scale pilots (LSP) in the building energy management domain. The developed ontology is used in MATRYCS platform [79] to manage the building energy domain data. The figure shows that the coverage of the dataset for LSP7 is the lowest at 78%, but this does not mean that the classification in the ontology is not suitable for the classification of the dataset. The data in LSP7 is related to energy consumption. The data

TABLE 3. Results of coverage.

Pilot	Datasets	Coverage(%)
LSP1	heating, electricity, water, Working Time, SCADA, weather data, solar power plant, calendar	97
LSP2	energy consumption data	87
LSP3	historical data from buildings	100
LSP4	BEMS, Changing station, Photovoltaic, Second Life Battery Storage	87
LSP5	reading from meter, solar projects production, EPC, Electric appliances, Residential solar production, Location of solar projects, solar projects general info, Weather information, Billing data period distribution, billing information	81
LSP6	Historical data, monitoring data	100
LSP7	Consumption data	78
LSP8	data for public buildings, High School11-9, Kindergarten, air quality recorders	80
LSP9	Certificados	100
LSP10	Electric meters, Heat meters with Metbox, Electronic Monitoring System, city district heating	87
LSP11	PORTUGALETE, Riberas, gas, Datos	84

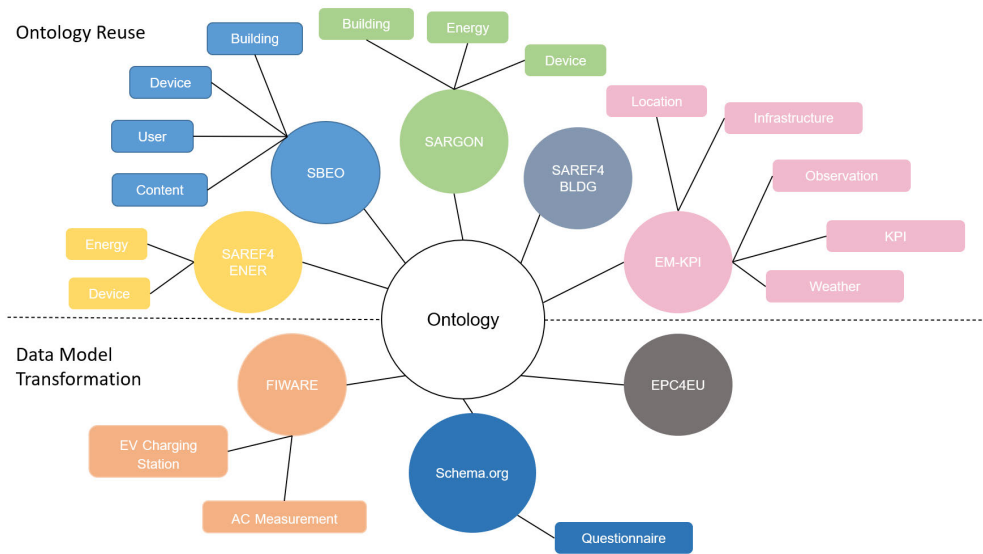


FIGURE 4. Ontology concept.

in LSP7 defined as uncovered are, for instance, *Fixed Tariff Term*, *Energy Pick*, and *Peak Hours*. These data can also be considered as subclasses of energy tariff, which is available in our ontology.

H. BUILDING ENERGY ONTOLOGY

As described in the previous section, we have defined the framework to develop an ontology that covers multiple domains. In this section, we display the final ontology to give a visual impression.

Figure 4 shows the top-level structure of the developed ontology and the importance of reusing both the existing ontology and data model. The final ontology is developed based on two knowledge sources: ontology and data model. Concepts and relationships from the existing ontologies are reused such as SAREF4ENER, SAREF4BLDG, SBE0, SARGON, and EM-KPI Ontology. The data model of Questionnaire, ACMeasurement, EPC, and EVChargingSatation are transformed in the Data Model Transformation step and integrated into this ontology. This shows how the developed ontology covers the required domain defined

in subsection IV-A for building energy data management. This structure is suggested by the Neon [10] as networked ontologies. The advantage of Networked Ontologies lies in their ability to address the limitations of single, monolithic ontologies by leveraging distributed knowledge representation in building energy management. This also shows that different ontologies may contain the same concept (e.g., device in SAREF4ENER and SBE0). Therefore, the ontology matching and merging step is important.

I. COMPARATIVE ASSESSMENT

In order to demonstrate the strengths and advantages of our approach in comparison to existing methodologies, we provide a comprehensive analysis that compares our method with the state-of-the-art methodology for multi-aspect ontology development proposed in [12].

The multi-aspect ontology development methodology comprises four stages:

- 1) Stage 1: Ontology Aspect Identification
- 2) Stage 2: Aspect Ontology Development

- 3) Stage 3: Aspect-Level Implementation of the Multi-Aspect Ontology
- 4) Stage 4: Global-Level Conceptualization Implementation

In Stage 1 of the multi-aspect ontology development methodology, our framework identifies the ontology's purpose and scope. The specification of requirements is achieved through competency questions. In Stage 2, the methodology also covers ontology reuse and conceptualization. However, it does not detail the process for choosing the appropriate ontology, requiring expert intervention for decision-making. Additionally, the transition from a data model to an ontology is not addressed, potentially increasing the manual workload involved in conceptualization. Stages 3 and 4 only involve manual ontology matching in the methodology proposed by [12]. The remaining steps closely align with the approach developed in our framework. In summary, our proposed methodology outlines a method for sourcing relevant knowledge and introduces a process for converting external knowledge sources into ontology, thereby enhancing the automation level of ontology development.

V. CONCLUSION

To determine the ontology development framework, we conducted a massive literature review and summarized previous ontology construction methods. Based on state-of-the-art techniques, we provided a semi-automatic ontology development framework, which can efficiently search the available ontologies and generate one ontology according to the requirements. Additionally, if the existing ontology is incomplete, we proposed a clear process based on existing tools and methods to convert the target data model into an ontology, thus saving time spent on constructing the ontology. Furthermore, we addressed the heterogeneity problem for building energy data management by developing a multi-domain ontology based on the proposed ontology development framework. We assessed the efficacy of the developed ontology through evaluation with 11 diverse data-intensive pilots.

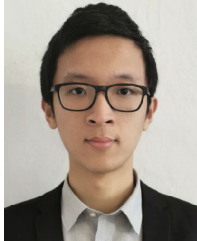
In the future, a tool, which can support the ontology development framework and contains all the necessary functions, can be developed. This will reduce the effort to manually combine tools for ontology matching, ontology merging and data model transformation. The developed building energy ontology can be demonstrated in practical applications to showcase the benefits of the developed ontology in a real-world building energy data management platform.

REFERENCES

- [1] R. Agrawal et al., "The Claremont report on database research," *SIGMOD Rec.*, vol. 37, pp. 9–19, Jun. 2008.
- [2] K. I. Kotis, G. A. Vouros, and D. Spiliotopoulos, "Ontology engineering methodologies for the evolution of living and reused ontologies: Status, trends, findings and recommendations," *Knowl. Eng. Rev.*, vol. 35, p. e4, Jan. 2020.
- [3] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquisition*, vol. 5, no. 2, pp. 199–220, Jun. 1993.
- [4] T. Tudorache, "Ontology engineering: Current state, challenges, and future directions," *Semantic Web*, vol. 11, no. 1, pp. 125–138, Jan. 2020.
- [5] M. Uschold and M. King, *Towards a Methodology for Building Ontologies*. Citeseer, 1995.
- [6] M. Fernández-López, A. Gómez-Pérez, and N. Juristo, "Methontology: From ontological art towards ontological engineering," in *Proc. Ontol. Eng. AAI Spring Symp. Ser.* Stanford, CA, USA: Stanford Univ., EEUU, Mar. 1997.
- [7] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford, CA, USA, Stanford Univ., Tech. Rep., 2001.
- [8] Y. Sure, S. Staab, and R. Studer, "On-to-knowledge methodology (OTKM)," in *Handbook on Ontologies*. Cham, Switzerland: Springer, 2004, pp. 117–132.
- [9] R. Gacitua, P. Sawyer, S. Piao, and P. Rayson, "Ontology acquisition process: A framework for experimenting with different NLP techniques," in *Proc. U.K. E-Sci. All Hands Conf.*, 2007, pp. 561–567.
- [10] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López, "The NeOn methodology framework: A scenario-based methodology for ontology development," *Appl. Ontol.*, vol. 10, no. 2, pp. 107–145, Sep. 2015.
- [11] A. Abdelghany, N. Darwish, and H. Hefni, "An agile methodology for ontology development," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 2, pp. 170–181, Apr. 2019.
- [12] A. Smirnov, T. Levashova, A. Ponomarev, and N. Shilov, "Methodology for multi-aspect ontology development," in *Proc. Int. Conf. Decis. Support Syst. Technol.* Cham, Switzerland: Springer, 2021, pp. 97–109.
- [13] A. Gyrard, A. Kung, O. Genest, and A. Moreau, "SAREF-compliant knowledge discovery for semantic energy and grid interoperability," in *Proc. IEEE 7th World Forum Internet Things (WF-IoT)*, Jun. 2021, pp. 71–76.
- [14] M. Poveda-Villalón and R. Garcia-Castro, "Extending the SAREF ontology for building devices and topology," in *Proc. 6th Linked Data Archit. Construct. Workshop*, vol. 2159, 2018, pp. 16–23.
- [15] Q. Khalid, A. Fernandez, M. Lujak, and A. Doniec, "SBEO: Smart building evacuation ontology," *Comput. Sci. Inf. Syst.*, vol. 20, p. 46, 2022.
- [16] M. Haghgooy, I. Sychev, A. Monti, and F. H. P. Fitzek, "SARGON—Smart energy domain ontology," *IET Smart Cities*, vol. 2, no. 4, pp. 191–198, Dec. 2020.
- [17] Y. Li, R. García-Castro, N. Mihindukulasooriya, J. O'Donnell, and S. Vega-Sánchez, "Enhancing energy management at district and building levels via an EM-KPI ontology," *Autom. Construct.*, vol. 99, pp. 152–167, Mar. 2019.
- [18] *Fiware Smart-Data-Models*. Accessed: Feb. 23, 2023. [Online]. Available: <https://www.fiware.org/smart-data-models/>
- [19] R. V. Guha, D. Brickley, and S. Macbeth, "Schema.ORG: Evolution of structured data on the web," *Commun. ACM*, vol. 59, no. 2, pp. 44–51, 2016.
- [20] J. R. Centre, F. Vinci, V. Serna-González, C. Valmaseda, F. Miguel-Herrero, F. Pignatelli, G. Martirano, and G. Hernandez-Moral, "Harmonisation of datasets of energy performance certificates of buildings across Europe: ELISE energy and location applications," Publications Office of the EU, Luxembourg, U.K., Tech. Rep., 2021, doi: 10.2760/500135.
- [21] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data Knowl. Eng.*, vol. 25, nos. 1–2, pp. 161–197, Mar. 1998.
- [22] P. R. Smith and R. Sarfaty, "Creating a strategic plan for configuration management using computer aided software engineering (case) tools," EG and G Idaho, Inc., Idaho Falls, ID, USA, Tech. Rep. 10160331, 1993.
- [23] N. Guarino and P. Giaretta, "Ontologies and knowledge bases," in *Towards Very Large Knowledge Bases*. Amsterdam, The Netherlands: IOS Press, 1995, pp. 1–2.
- [24] P. Spyns, R. Meersman, and M. Jarrar, "Data modelling versus ontology engineering," *ACM SIGMOD Rec.*, vol. 31, no. 4, pp. 12–17, Dec. 2002.
- [25] S. Abburu and S. B. Golla, "Ontology storage models and tools: An authentic survey," *J. Intell. Syst.*, vol. 25, no. 4, pp. 539–553, Oct. 2016.
- [26] J. Euzenat, T. Le Bach, J. Barrasa, P. Bouquet, J. De Bo, R. Dieng, M. Ehrig, M. Hauswirth, M. Jarrar, and R. Lara, "State of the art on ontology alignment," *Knowl. Web Deliverable D*, vol. 2, pp. 2–3, 2004.
- [27] D. L. McGuinness and F. Van Harmelen, "Owl web ontology language overview," *W3C Recommendation*, vol. 10, no. 10, p. 2004, 2004.

- [28] G. Pardo-Castellote, "OMG data-distribution service: Architectural overview," in *Proc. 23rd Int. Conf. Distrib. Comput. Syst. Workshops*, May 2003, pp. 200–206.
- [29] L. Audibert, "UML 2," Département Informatique, Institut Universitaire de Technologie de Villetaneuse, Villetaneuse, France, Tech. Rep., 2007.
- [30] M. C. Suárez-Figueroa, A. Gómez-Pérez, and B. Villazón-Terrazas, "How to write and use the ontology requirements specification document," in *Proc. OTM Confederated Int. Conf. Meaningful Internet Syst.* Cham, Switzerland: Springer, 2009, pp. 966–982.
- [31] H. A. Carneiro and E. Mylonakis, "Google Trends: A we-based tool for real-time surveillance of disease outbreaks," *Clin. Infectious Diseases*, vol. 49, no. 10, pp. 1557–1564, Nov. 2009.
- [32] N. Saienko and Y. Lavrysh, "Information assessment skills development through the mobile learning strategies for bachelors at technical university," *Dilemas Contemporáneos, Educación, Política Valores*, vol. 7, no. 2, pp. 1–22, 2020.
- [33] J. I. Jerkovic, *SEO Warrior: Essential Techniques for Increasing Web Visibility*. Sebastopol, CA, USA: O'Reilly Media, 2009.
- [34] A. Hogan, "Knowledge graphs," *ACM Comput. Surv.*, vol. 54, no. 4, pp. 1–37, 2021.
- [35] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "FreeBase: A collaboratively created graph database for structuring human knowledge," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2008, pp. 1247–1250.
- [36] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus for a web of open data," in *The Semantic Web*. Cham, Switzerland: Springer, 2007, pp. 722–735.
- [37] M. Färber, F. Bartscherer, C. Menne, and A. Rettinger, "Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO," *Semantic Web*, vol. 9, no. 1, pp. 77–129, Nov. 2017.
- [38] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Commun. ACM*, vol. 57, no. 10, pp. 78–85, Sep. 2014.
- [39] P.-Y. Vandenbussche, G. A. Atemezding, M. Poveda-Villalón, and B. Vatant, "Linked open vocabularies (LOV): A gateway to reusable semantic vocabularies on the web," *Semantic Web*, vol. 8, no. 3, pp. 437–452, Dec. 2016.
- [40] V. Bazjanac and D. B. Crawley, "Industry foundation classes and interoperable commercial software in support of design of energy-efficient buildings," in *Proc. Building Simulation*, vol. 2. Boston, MA, USA: Addison-Wesley, 1999, pp. 661–667.
- [41] E. P. Bontas, M. Mochol, and R. Tolksdorf, "Case studies on ontology reuse," in *Proc. Int. Conf. Knowl. Manage.*, 2005, vol. 74, pp. 345–353.
- [42] E. G. Calderola and A. M. Rinaldi, "An approach to ontology integration for ontology reuse," in *Proc. IEEE 17th Int. Conf. Inf. Reuse Integr. (IRI)*, Jul. 2016, pp. 384–393.
- [43] D. C. Bulterman, L. F. G. Soares, and M. G. C. Pimentel, "Proceedings of the eighth ACM symposium on document engineering: DOCENG'08," in *Proceedings*. New York, NY, USA: Association for Computing Machinery, 2008.
- [44] Y. Li, Z. A. Bandar, and D. McLean, "An approach for measuring semantic similarity between words using multiple information sources," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 4, pp. 871–882, Jul. 2003.
- [45] S. Chávez-Feria, R. García-Castro, and M. Poveda-Villalón, "Converting UML-based ontology conceptualizations to OWL with Chowik," in *Proc. Extended Semantic Web Conf.*, 2021, pp. 44–48.
- [46] P. Shvaiko and J. Euzenat, "Ontology matching: State of the art and future challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 158–176, Jan. 2013.
- [47] 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, Nov. 2022.
- [48] K. Brown, *Encyclopedia of Language and Linguistics*, vol. 1. Amsterdam, The Netherlands: Elsevier, 2005.
- [49] S. Massmann, S. Raunich, D. Aumüller, P. Arnold, and E. Rahm, "Evolution of the coma match system," *Ontol. Matching*, vol. 49, pp. 49–60, Oct. 2011.
- [50] X. Su and J. A. Gulla, "Semantic enrichment for ontology mapping," in *Proc. Int. Conf. Appl. Natural Lang. Inf. Syst.* Cham, Switzerland: Springer, 2004, pp. 217–228.
- [51] M. Cristani and R. Cuel, "A survey on ontology creation methodologies," *Int. J. Semantic Web Inf. Syst.*, vol. 1, no. 2, pp. 49–69, 2005.
- [52] D. L. Rubin, N. F. Noy, and M. A. Musen, "Protégé: A tool for managing and using terminology in radiology applications," *J. Digit. Imag.*, vol. 20, no. S1, pp. 34–46, Nov. 2007.
- [53] P. Hitzler, M. Krötzsch, M. Ehrig, and Y. Sure, "What is ontology merging?" in *American Association for Artificial Intelligence*. Washington, DC, USA: AAAI Press, 2005, p. 4.
- [54] I. Osman, S. Ben Yahia, and G. Diallo, "Ontology integration: Approaches and challenging issues," *Inf. Fusion*, vol. 71, pp. 38–63, Jul. 2021.
- [55] D. Geleta, A. Nikolov, M. O. Donoghue, B. Rozemberczki, A. Gogleva, V. Tamma, and T. R. Payne, "Ontomerger: An ontology integration library for deduplicating and connecting knowledge graph nodes," 2022, *arXiv:2206.02238*.
- [56] N. F. Noy and M. A. Musen, "The PROMPT suite: Interactive tools for ontology merging and mapping," *Int. J. Hum.-Comput. Stud.*, vol. 59, no. 6, pp. 983–1024, Dec. 2003.
- [57] S. Raunich and E. Rahm, "ATOM: Automatic target-driven ontology merging," in *Proc. IEEE 27th Int. Conf. Data Eng.*, Apr. 2011, pp. 1276–1279.
- [58] S. Tartir, I. B. Arpinar, M. Moore, A. P. Sheth, and B. Aleman-Meza, "OntoQA: Metric-based ontology quality analysis," Houston, TX, USA, Tech. Rep., 2005.
- [59] Ó. Corcho, A. Gómez-Pérez, R. González-Cabero, and M. C. Suárez-Figueroa, "ODEval: A tool for evaluating RDF (S), DAML+ OIL, and OWL concept taxonomies," in *Proc. IFIP Int. Conf. Artif. Intell. Appl. Innov.* Springer, 2004, pp. 369–382.
- [60] M. Fernández, I. Cantador, and P. Castells, "CORE: A tool for collaborative ontology reuse and evaluation," in *Proc. EON@ WWW*, 2006, pp. 1–9.
- [61] M. Poveda-Villalón, A. Gómez-Pérez, and M. C. Suárez-Figueroa, "OOPS! (Ontology pitfall scanner!): An on-line tool for ontology evaluation," *Int. J. Semantic Web Inf. Syst.*, vol. 10, no. 2, pp. 7–34, Apr. 2014.
- [62] D. Garijo, "WIDOCO: A wizard for documenting ontologies," in *Proc. Int. Semantic Web Conf.* Cham, Switzerland: Springer, 2017, pp. 94–102.
- [63] A. Alobaid, D. Garijo, M. Poveda-Villalón, I. Santana-Perez, and O. Corcho, "Ontology, a tool for collaborative development of ontologies," in *Proc. ICBO*, 2015, pp. 1–2.
- [64] T. G. Stavropoulos, D. Vrakas, D. Vlachava, and N. Bassiliades, "BOnSAI: A smart building ontology for ambient intelligence," in *Proc. 2nd Int. Conf. Web Intell., Mining Semantics*, Jun. 2012, pp. 1–12.
- [65] N. Shah, K.-M. Chao, T. Zlamanic, and A. Matei, "Ontology for home energy management domain," in *Proc. Int. Conf. Digit. Inf. Commun. Technol. Appl.*, vol. 167, 2011, pp. 337–347.
- [66] M. H. Rasmussen, M. Lefrançois, G. F. Schneider, and P. Pauwels, "BOT: The building topology ontology of the W3C linked building data group," *Semantic Web*, vol. 12, no. 1, pp. 143–161, Nov. 2020.
- [67] L. Liu, P. Hagedorn, and M. König, "An ontology integrating as-built information for infrastructure asset management using BIM and semantic web," in *Proc. Eur. Conf. Comput. Construct.*, Jul. 2021, pp. 99–106.
- [68] W. Terkaj and A. Šojić, "Ontology-based representation of IFC EXPRESS rules: An enhancement of the ifcOWL ontology," *Autom. Construct.*, vol. 57, pp. 188–201, Sep. 2015.
- [69] Q. Khalid, A. Fernandez, M. Lujak, and A. Doniec, "SBEO: Smart building evacuation ontology," *Comput. Sci. Inf. Syst.*, vol. 20, no. 1, pp. 51–76, 2023.
- [70] N. Bountouni, F. Lampathaki, S. Kousouris, A. Tsitsanis, G. Vafeiadis, and D. Vergeti, "The BIMERR interoperability framework: Towards BIM enabled interoperability in the construction sector," in *Proc. 38th Int. Symp. Autom. Robot. Construct. (ISARC)*, Nov. 2021, pp. 94–101.
- [71] SAREF4ENER: An Extension of SAREF for the Energy Domain Created in Collaboration With Energy@ Home and EEBus Associations. Accessed: Feb. 23, 2023. [Online]. Available: <https://saref.etsi.org/saref4ener/v1.1.2/>
- [72] J. Verhoosel, D. Rothengatter, F. Rumph, and M. Konsman, "An ontology for modeling flexibility in smart grid energy management," in *Proc. Eur. Conf. Product Process Modelling—eWork eBus. Archit., Eng. Construct. (ECCPPM)*, Reykjavík, Iceland, Jul. 2012, pp. 931–993.
- [73] V. Corrado, I. Ballarini, L. Madrazo, and G. Nemirovskij, "Data structuring for the ontological modelling of urban energy systems: The experience of the SEMANCO project," *Sustain. Cities Soc.*, vol. 14, pp. 223–235, Feb. 2015.
- [74] S. A. Moschen, J. Macke, S. Bebbler, and M. B. C. da Silva, "Sustainable development of communities: ISO 37120 and UN goals," *Int. J. Sustainability Higher Educ.*, vol. 20, no. 5, pp. 887–900, Jul. 2019.
- [75] V. Kukkonen, A. Kucikavci, M. Seidenschur, M. H. Rasmussen, K. M. Smith, and C. A. Hvuid, "An ontology to support flow system descriptions from design to operation of buildings," *Autom. Construct.*, vol. 134, Feb. 2022, Art. no. 104067.

- [76] F. Amarger, S. Bernard, N. Hernandez, and C. Roussey, "Use of SOSA ontology to describe agricultural observations: 2 use cases of plot observations," *Tech. Rep.*, 2017.
- [77] R. Iannella and J. McKinney, "vCard ontology-for describing people and organizations," W3C Group, *Tech. Rep.*, NOTE-vcard-rdf-20140522, 2014.
- [78] J. R. Hobbs and F. Pan, "Time ontology in OWL," *W3C Working Draft*, vol. 27, no. 133, pp. 3–36, 2006.
- [79] M. Pau, P. Kapsalis, Z. Pan, G. Korbakis, D. Pellegrino, and A. Monti, "MATRYCS—A big data architecture for advanced services in the building domain," *Energies*, vol. 15, no. 7, p. 2568, 2022.



ZHIYU PAN (Member, IEEE) received the M.Sc. degree in automatic engineering from RWTH Aachen University, in 2020, where he is currently pursuing the Ph.D. degree. He is leading the Research Group, Institute for Automation of Complex Power Systems, RWTH Aachen University. His research interests include semantic similarity and big data management in the energy domain.



YUTING GAO received the bachelor's degree in industrial engineering from Hefei University and the University of Applied Sciences and Art (HAWK). She is currently pursuing the master's degree in sustainable energy supply with RWTH Aachen University.



FERDINANDA PONCI (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Politecnico di Milano, Italy, in 2002. In 2003, she joined the Department of Electrical Engineering, University of South Carolina, Columbia, SC, USA, as an Assistant Professor, and was tenured and promoted, in 2008. In 2009, she joined the Institute for Automation of Complex Power Systems, E.ON Research Center, RWTH Aachen University, Aachen, Germany, where she is currently a Professor for monitoring and distributed control for power systems.



ANTONELLO MONTI (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Politecnico di Milano, Italy, in 1994. In 1995, he started the career with Ansaldo Industria and then moved to Politecnico di Milano as an Assistant Professor. In 2000, he joined the Department of Electrical Engineering, University of South Carolina, Columbia, SC, USA, as an Associate Professor, and then a Full Professor. Since 2008, he has been the Director of the Institute for Automation of Complex Power Systems, E.ON Energy Research Center, RWTH Aachen University, Germany. Since 2019, he has been with the Center for Digital Energy Aachen, Fraunhofer FIT.

...