

GOTTFRIED WILHELM LEIBNIZ UNIVERSITÄT HANNOVER

Masters thesis

# Comparison of Ontologies in Energy Research

submitted by

*Alexandro Steinert*

Matriculation Number: 10014754

Advisor

First Examiner

Second Examiner

Stephan Ferenz

Prof. Dr. Sören Auer

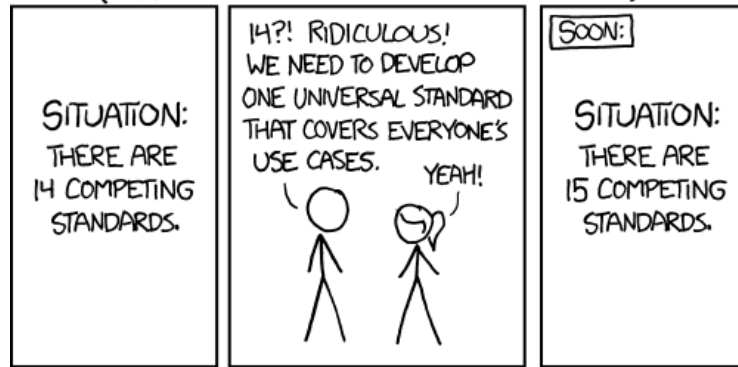
Prof. Dr.-Ing. Astrid Nieße



May 22, 2023



HOW STANDARDS PROLIFERATE:  
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)



How Standards Proliferate Comic made by xkcd [66].



## **Abstract**

With the growing threat of the climate crisis, energy is a more prevalent topic than ever and as such energy research is increasingly important. Ontologies as a concept are an important tool in collecting, mapping and reusing data in research. Furthermore, they are used to make data as well as metadata better machine interpretable. Since there are multiple ontologies available a researcher has to choose which one to use in his projects. While there are some comparisons of ontologies in energy research. There is no comparison of ontology metadata.

As such this thesis aims to compare ontologies and their metadata to support researchers in choosing an ontology for their energy research project. Furthermore, a software tool to assist researchers in picking an ontology is developed in this thesis.

Therefore, ontologies that are in use in energy research are identified by using a modified Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) approach. Through this, eight usable ontologies are identified. These eight ontologies are then compared in four categories and a total of 21 criteria. The criteria were developed in this thesis for the comparison. Each category is scored to simplify the comparison. As the best ontology to use depends on the use case, there can not be a definitive answer to the question of which ontology is most suitable. However, this thesis helps in choosing ontologies, by giving an easy to understand score for quick selection as well as a detailed comparison for aiding in in-depth decision making. The ontology with the highest accumulated score is the Open Energy Ontology (OEO).

Afterwards a tool that can be integrated into existing ontology databases is developed. This tool makes use of the collected metadata to help enhance the search in existing databases.

This thesis shows that comparing ontology metadata is useful, as there are many differences in ontologies apart from their content. Furthermore, it shows that an integration of the criteria in ontology databases is useful in aiding researchers. As such it is important to compare ontologies and aid in choosing them by showing their metadata differences.

**Keywords:** Energy, Ontology, Semantic Web, Metadata, Comparison, FAIR



# Contents

<b>1. Introduction</b>	<b>1</b>
1.1. Research Question	1
1.2. Thesis Structure	1
<b>2. Fundamentals</b>	<b>3</b>
2.1. Semantic Web	3
2.2. Metadata	3
2.3. Ontologies	4
2.3.1. Ontology Languages	4
2.3.2. Existing Software	10
<b>3. Ontologies in Energy Research</b>	<b>15</b>
3.1. Method and Selection	15
3.2. Selected ontologies	18
<b>4. Analysis of Ontologies</b>	<b>21</b>
4.1. Computing a Score	21
4.2. Criteria for Analysis of Ontologies	21
4.2.1. Related Work	21
4.2.2. Criteria Categories	22
4.2.3. Best Practice	22
4.2.4. Practical Implementation	23
4.2.5. Maintenance and Accessibility	24
4.2.6. Governance	25
4.2.7. Python Script for Data Compilation	25
4.3. Analysis of the Ontologies	25
4.3.1. Issues during Data Collection	25
4.3.2. Best practice	26
4.3.3. Practical Implementation	27
4.3.4. Maintenance and Accessibility	28
4.3.5. Governance	30
4.4. Score Evaluation	31
<b>5. Development of an Ontology Picking Tool</b>	<b>33</b>
5.1. Existing Tools	33
5.2. Planned Tool	33
5.2.1. Problems with Existing Tools	33
5.2.2. Planning	34
5.3. Development	34
5.3.1. Possible Further Development	35
5.4. Exemplary Integration in an Existing Database	35
5.4.1. Demonstrative Integration	35
5.4.2. Integration Challenges	36
5.4.3. Possible Further Development	37
<b>6. Conclusion</b>	<b>39</b>
6.1. Summary	39
6.2. Future Work	39
<b>A. Example Code for a Veterinarian Clinic</b>	<b>41</b>
<b>B. Tables with Data from the Analysis</b>	<b>45</b>
<b>A. References</b>	<b>III</b>
<b>B. List of Figures</b>	<b>VII</b>
<b>C. List of Tables</b>	<b>IX</b>

D. Declaration of Academic Integrity . . . . . **XI**



## 1. Introduction

The scientific field of energy research is more relevant and interesting than ever before since the threat of global warming is ever more present. Especially the accomplishments in the area of renewable energy and machine learning based energy systems, seem promising in regards to tackling the global warming crisis.

In energy, as well as in other fields of research, research data and software are an integral part [18]. As such there is an ever greater urgency to produce the research data as well as software openly available as to advance collaborative research and make it possible [42].

Wilkinson et al. [72] developed the FAIR principles as guidelines to aid in the responsible handling of research data. The goal of the principles is to enable open research work [72, 9]. FAIR is an acronym, which stands for the following:

1. **F**indable
2. **A**ccessible
3. **I**nteroperable
4. **R**eusable

A focus here is on machine readability as well as discoverability of data [72]. The principles were extended by Hasselbring et al. [23] to facilitate their use on as well as the development of research software [23].

To enable findable, accessible, interoperable and reusable (FAIR) research data and software in energy research it is important to ensure interoperability. For this cause it is important to define widely used terms. Ontologies are a concept from the semantic web that are used for this purpose. Using an ontology allows better machine findability as well as easier machine interpretability of data. Furthermore, they allow the integration of data from different datasets [3].

There are many ontologies available in energy research. All of them have a different focus, recognition and scope [71]. The ontologies are often available from different sources. All this makes it harder for researchers to choose the right ontology for their usecase.

To solve this issue, this thesis aims to methodically compare the information that is available on ontologies. Through this, the work allows for easier comparison between ontologies and their metadata, thus aiding researchers in selecting ontologies. To further ease the comparison between ontologies, a tool is developed to aid researchers in selecting an ontology. This is achieved by developing a tool that can be integrated into ontology databases to introduce more search criteria. In the following, the research questions that need to be answered to accomplish this, are formulated and the structure of the thesis is defined.

### 1.1. Research Question

The goal of this work is to compare ontologies in energy research. This work aims to aid in ordering ontologies according to criteria such as accessibility, maintenance and more. As such, this thesis aims to answer the research question:

**Which ontology is suitable for a project in energy research and how can researchers be supported in choosing one?**

To answer this question appropriately it is further divided into four parts:

1. Which ontologies are used in energy research?
2. How can ontologies be compared?
3. What are the relevant criteria for this comparison?
4. How can researchers be supported in choosing an ontology?

### 1.2. Thesis Structure

This thesis is divided into eight chapters including the introduction. The fundamentals, section 2, explains the necessary concepts for understanding this thesis with a focus on the semantic web, metadata as well as ontologies. After establishing a common ground, section 3 outlines the method used to find ontologies applicable to energy research as well as the method to define a subset of ontologies to compare thus answering the first research question defined in subsection 1.1. Following is the analysis, section 4, which defines the comparison method and presents the results of the comparison thus answering research questions two and three. The knowledge gathered in the previous section is applied in the development of an ontology picking tool in section 5. This section shows the process

of developing a tool that supports researchers in choosing an ontology and answers research question number four. The conclusion, section 6, sums up the result of this work and points out possible future work that can be done.

## 2. Fundamentals

This chapter covers the scientific background needed for the remaining thesis. The following sections provide a rough overview on the overlying topic. It most notably presents the essentials of semantic web and ontologies.

### 2.1. Semantic Web

This subsection presents a short overview of the term semantic web and its technologies. The semantic web is conceived and developed by the World Wide Web Consortium (W3C) and can be understood as a concept in computer science. It is as an alternative approach to the web with a focus on data description instead of web document layout. The goal of this approach is to make the web more suited for computers such that computers may support humans better on the web. To achieve this goal, the data on the web needs to be more machine-readable and accessible. The idea behind propagating the semantic web is not to replace today's web but to transform it into the semantic web step by step. There are different technologies in development and in use to support this transformation. One of which is the development of ontologies as understood by computer science [3].

### 2.2. Metadata

This section provides a short overview on the term metadata and how it is intrinsically connected with ontologies and the semantic web.

Metadata in its most basic form is generally understood as data about data or information about information. A very basic example of this is the creation date, which is automatically created and stored for every file on a computer. It has a deep history, that is, at least, going back to the documentation of data available in museums and libraries.

Metadata can be divided into different types. This thesis will follow the main definitions established by Zeng and Qin [74] on the basis of the metadata division by the National Information Standards Organization (NISO) [61].

A coarse overview of metadata categories can be found below:

*structural metadata* data about the relations between different parts of information, e.g. data about the order of photos in a slideshow.

*administrative metadata* data describing administrative rights of/to the information, as well as other administrative data about the information, e.g. the access right of an employee in a given software

*descriptive metadata* data for finding or describing the information, e.g. the author of a book

*meta metadata* data about metadata

There exists many different standards for recording metadata. Metadata can be stored either internally, alongside the data that it is about, or externally, meaning it is stored in separate files or even physically separated. Metadata itself is a mapping between a property of an information entity and a value, also called a statement. A description is a collection (more than one) of statements about an information entity. An aggregation of metadata is called a record. Metadata can be gathered by hand, for example when a person is writing where a photograph is taken on the back of the photograph. Automatically, which means the data is gathered by a computer, which happens, for example, when creating a file and the creation date is saved automatically. Furthermore, metadata can be created semi-automatically,

Metadata that is aggregated for a specific purpose is called a metadata element set. This aggregation can be divided into two different groups depending on the structure. A set can have a flat structure, meaning all of its elements are arranged linearly or in a nested structure, when there exist relationships between the elements. An example of a flat structured metadata element set would be the Dublin Core. It is an element set consisting of fifteen elements for describing resources and also a standard under International Organization for Standardization (ISO). An example for a nested element set can be seen in <https://schema.org/>. Furthermore, there is a distinction between a set defined by its maintainer, called a namespace schema and set that can borrow elements from another set called application profile schema. An element set together with the encoding of the set using a standard language is often called a metadata schema or metadata standard.

Ontologies can be used in two ways. They can be used in tandem with another metadata schema and can restrict the metadata schema by only allowing a limited number of terms or as a guidance for

the schema. Ontologies can also be used as metadata schemas themselves. Ontologies are schemas that are stored externally, either as namespaces or application profile schemas and have a nested structure [74, 61].

## 2.3. Ontologies

An Ontology could be intuitively described as a representation of knowledge in form of a curated and structured collection. It consists of a finite collection of terms, their definitions and the relationships between them. The most common relationship between those terms is hierarchical, however, unlike an taxonomic hierarchy, where the only relation between terms is of hierarchical nature, there can be other relationships defined [3].

Ontologies are a very important tool for semantic knowledge management as they enable key concepts such as:

- Collection
- Mapping
- and Reuse of knowledge

Furthermore, it is possible to make inferences between the knowledge in ontologies. This can either be done automatically or by hand. Ontologies allow merging data from different institutions by either mapping different terminology for the same concept and as such being able to merge the data for that concept, or by clarifying that the term used means the same for both institutions. To give an example: if two different scientific institutions work on a shared paper repository, the use of ontologies would enable a scientist to query the repository and find papers related to his query that might not use the same terminology, but describe the same concept. Enabling him to identify, if a problem he is working on is already undergoing scientific discussion in the partner institution even if it is referenced by a different term.

Ontologies are usually distinguished into two different types:

*Upper ontologies:* This ontology type describes a very general concept using terms that are not exclusive to any domain of knowledge.

*Domain ontologies:* This ontology type describes a concept in a specific domain of knowledge, e.g. energy.

It is common practice to form a new ontology by using parts of existing ontologies or creating more specific domain ontologies based on upper ontologies, enabling the reuse of knowledge [3, 49]. There are a number of different upper ontologies as well as approaches to building ontologies. One example of this is the Open Energy Ontology (OEO), which is based on the Basic Formal Ontology. It also uses the basic design principles of the Basic Formal Ontology [7].

As ontologies describe relationship between terms in a logical way, it is possible to form logical conclusion between them, as well as inferring these conclusions using computer algorithms called Reasoners, that work based on first order logic. They can be built using the open- or closed-world-assumption, which means that either assumptions that are not known to be true are false in case of the closed-world-assumption or, in case of the open-world-assumption, meaning that these assumptions cannot be known and as such no statement about truth or falsehood can be made. Most ontologies operate under the open-world-assumption. Herein lies a key difference with other knowledge representations such as databases, as those traditionally support the closed-world-assumption. Another key difference to other means of storing data, like databases is that there can be instances of a term, they are however, not unique and need to be assigned a unique identifier to be properly distinguished.

It is important to mention that there is no such thing as a inherently right or wrong ontology content wise, as there can be multiple different representations of knowledge. However, for the sake of collaboration it is important to try to reuse ontologies when possible [3, 4, 7].

Ontologies are a concept for knowledge representation and there is no definitive language to record them. As such there are a few different options, some of which will be highlighted in the section below.

### 2.3.1. Ontology Languages

Any language capable of expressing first order logic can describe an ontology adequately. Graph representations where terms are represented by nodes and the edge relation models the semantic relation of these terms, could also be employed. There are however some commonly used languages that are de facto standards, which will be highlighted in the following paragraphs. The first language that will be focused on is the extensible markup language (XML) which is a markup language with a focus on metadata that is used as a base for the following languages. After that Resource Description

Framework (RDF)/Resource Description Framework Schema (RDFS) will be explained, which forms a very powerful language pair for describing ontologies that can be expressed using XML. The last language that is explained is Web Ontology Language (OWL), which is a standard widely used today and based on RDF/RDFS [57, 49, 17].

**XML** The first language that will be highlighted is XML. It is not strictly a language used to describe ontologies. XML is detailed since it is used as a basis for those ontology languages used today, as well as many semantic web technologies. XML is a markup language like hypertext markup language (HTML). It is developed by W3C and recommended by them for use in data description.

Both languages are derivatives of the standard generalized markup language (SGML), which is in turn an ISO standard for text and information representation. XML differs in some key ways from HTML. To highlight those differences Figure 1 shows an example, showing veterinarian information formatted using both languages.

<pre> 1 &lt;h1&gt;Help Pets Inc.&lt;/h1&gt; 2 &lt;h2&gt;Contact us:&lt;/h2&gt; 3 &lt;p&gt;Jane Doe&lt;/p&gt; 4 &lt;p&gt;Tel.: +1 505-644-1287&lt;/p&gt; 5 &lt;p&gt;Mail: Jane.Doe@helppetsinc.com&lt;/p&gt; </pre>	<pre> 1 &lt;firm&gt; 2   Help Pets Inc. 3   &lt;veterinarian&gt; 4     Jane Doe 5     &lt;telephone_number&gt;+1 6       505-644-1287&lt;/telephone_number&gt; 7     &lt;email&gt;Jane.Doe@helppetsinc.com&lt; 8     /email&gt; 9   &lt;/veterinarian&gt; 10 &lt;/firm&gt; </pre>
--	---

Figure 1: Code snippet formatting veterinarian contact information in HTML (left) and XML (right).

It should be mentioned that both languages allow the programmer to define the layout and use of tags. However, as can be seen in the example above the focus of HTML lies on information formatting and in itself specifies no metadata for the displayed information. XML, however, focuses on attaching metadata to information. Tags can be defined by the user depending on the metadata needed and every tag consists of an opening and closing bracket [3].

As XML allows the user to attach metadata, it enables machines to consider the meaning of the data as well as making it better machine readable than HTML. As of now, it should be clear that HTML and XML serve two different purposes, namely describing the layout in which to present the information in the case of HTML and describing the given information further in the case of XML. The problem of HTML not being machine readable is actively worked on, as can be seen by the fact that HTML is updated to add more metadata functionality. The newest version of HTML, HTML5, as well as prior versions like XHTML, add metadata functionality by using RDFa or microdata to link certain used tags to metadata schemas/ontologies, like <https://schema.org/>. <https://schema.org/> is a metadata schema founded by a number of big web companies and contains a list of entities and relations between them used by websites to add metadata information.

With XML it is also possible to define namespaces with the keyword *xmlns*. This allows the reuse of different definitions between documents and the ability to distinguish between definitions of the same terms in different namespaces. There exists a query language for XML which enables it to be queried, not unlike databases, called xquery [3, pp. 25-59][74][17].

**RDF and RDFS** The next important language is called RDF. Used in conjunction with RDFS, it can be used to describe ontologies. The word language might be a bit misleading in this case, as the framework does not specify any syntax and works more in line of a model. It is however, commonly used together with XML as syntax. Other languages used as syntax for RDF and RDFS are for example JavaScript Object Notation for Linked Data (JSON-LD), Notation3 as well as Terse RDF Triple Language (Turtle) and N-Triples.

RDF defines statements. Those statement consist of a triplet of subject, predicate and object, which can also be understood as the logical formula  $P(s,o)$ , wherein  $P$  is the predicate,  $s$  the subject and  $o$  the object. Predicates are constricted to two parameters. As with XML RDF can be expressed as a graph as its basic structure. Taking the example from Figure 1 and adapting it to the RDF/RDFS structure can be found in the Figure 2 below, where the veterinarian (subject) works at (predicate) the animal clinic (object).

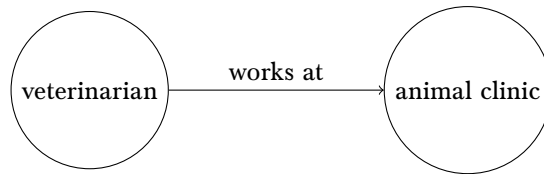


Figure 2: The example RDF statement `works_at(veterinarian, animal_clinic)` in graph form

Objects and subjects are represented as resources in RDF. Each resource has a International Resource Identifier (IRI), which is in most cases a Uniform Resource Locator (URL), or another kind of unique identifier. IRIs are a generalization of Uniform Resource Identifiers (URIs) with support for more languages. There are also literals in RDF which are basic datatypes. The binary statement of a predicate relating a subject to an object  $P(s,o)$  is found exactly like this in RDF/RDFS code. However, predicates are called properties and subject as well as object can either be a resource or a literal value. The statements itself in RDF code consist of the triple of a resource, a property and a value  $P(r,v)$ , where value can be a resource or a literal. The example in Figure 2 would look like the following triple in RDF:

*(<http://example.toy#animal-clinic>, <http://www.mydomain.org/works-at>, <http://example.name#veterinarian>)*

The following Figure 3 shows the example in RDF notation with XML syntax.

```

1 <rdf:RDF
2   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
4   xmlns:petclinic="http://www.mydomain.org/pet-clinic#">
5
6 <rdf:Description rdf:about="http://www.mydomain.org/pet-clinic#12345">
7   <rdf:type rdf:resource="&petclinic;veterinarian"/>
8   <petclinic:name>Jane Doe</petclinic:name>
9   <petclinic:email>Jane.Doe@helppetsinc.com</petclinic:email>
10  <petclinic:telephone>+1505 -644 -1287</petclinic:telephone>
11  <petclinic:worksat> rdf:resource="http://www.mydomain.org/pet-clinic#clinic1"/>
12 </rdf:Description>
13
14 <rdf:Description rdf:about="http://www.mydomain.org/pet-clinic#clinic1">
15   <rdf:type rdf:resource="&petclinic;clinic"/>
16   <petclinic:clinicName>Help Pets Inc.</petclinic:clinicName>
17 </rdf:Description>

```

Figure 3: Code snippet formatting the graph from Figure 2 in RDF using XML.

As can be seen in the above example the RDF links to the XML schema used as well as the RDF syntax. This is considered standard practice for any RDF document. The practice of using namespaces can be seen here again, as the terms name and title are part of the petclinic namespace. Namespaces enable the reuse of already defined terminologies in RDFS. The use of multiple namespaces is also possible

RDFS can be considered an extension of RDF. It is needed when using RDF to describe an ontology. On its own RDF only defines how to formulate statements and describe resources. It gives no information about the domain of knowledge. With RDFS exactly this domain knowledge is provided by defining the terminology of a given domain. To describe domains, RDFS enables the user to define classes, inheritance and properties similarly to object-oriented programming languages. A key difference to those terms in object-oriented programming languages is that properties are globally defined. Another key feature of the schema is that it enables the user to define restrictions for statements. For example it is possible to restrict the works at relation in the Figure 2 to a defined veterinarian class.

Using RDF and RDFS together it is now possible to describe statements within a domain and define the domain. As such, both used in conjunction provide a foundation for metadata representation. There exist multiple query languages for RDF/RDFS that work similarly to database query languages and can be used to extract information from a given RDF graph. The language recommended by the W3C is SPARQL Protocol And RDF Query Language (SPARQL) [3, pp. 65-109][57, 59].

```

1 <rdf:Property rdf:ID="worksat">
2   <rdfs:comment>
3     It relates only clinics to veterinarian.
4   </rdfs:comment>
5   <rdfs:domain rdf:resource="#clinic"/>
6   <rdfs:range rdf:resource="#veterinarian"/>
7 </rdf:Property>

```

Figure 4: Code Snippets restricting the works at relation from Figure 2 in RDFS using XML. Full code can be found in Appendix A

**OWL** While RDF/RDFS enables the user to describe domains, it is rather limited. For example predicates can only relate two concepts. It is also not possible to define hierarchies other than for subclasses and properties. OWL is a language developed and maintained by the W3C to address those issues. It is based on the language DAML+OIL, which was jointly developed by European and U.S. research groups [70]. There exist two different versions of OWL. The 2004 version called OWL1 in this document and the 2009 version called OWL2. As there are still ontologies written in OWL1 the following text will give a short overview of both flavors. The W3C recommends to use OWL2 when developing an ontology. Furthermore, it is a superset of OWL1 and as such ontologies written in OWL1 are compatible with OWL2. This section will first explain the basics of OWL1 and then explain the new concepts of OWL2 as well as other differences.

OWL1 is divided into three different sublanguages each with different levels of expressiveness, as well as computational requirements. It follows the core principles of RDF/RDFS as outlined above. With different levels of restrictions added according to the sublanguages [51]. A comparison between the sublanguages is shown in Table 1.

Table 1: Comparison of the OWL1 sublanguages.

OWL Lite	OWL DL	OWL Full
Least expressiveness	Middle expressiveness	Maximum expressiveness
Capable of expressing hierarchies, simple features and binary cardinality	Includes all OWL constructs but with restrictions	Full language
Least expressiveness	Computational complete and efficient reasoning	No guarantee for computational efficiency or completeness (as expected based on theoretical computer science)
Not fully compatible with RDF	Not fully compatible with RDF	Fully compatible with RDF
Subset of OWL DL	Subset of OWL Full	Full language

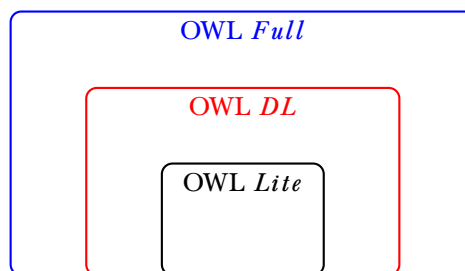


Figure 5: The relationship between the standard OWL1 sublanguages visualized.

The subset relationship of the different sublanguages is visualised in Figure 5. There exist different possible syntaxes for OWL1. However, one of the primary is RDF/XML. Since it is possible in RDF

to define/use different kinds of namespaces. As such at the root of an OWL1 ontology is a *rdf:RDF* element declaring every namespace in use. After this follows an *owl:Ontology* element in which basic assertions are declared for housekeeping. They, most importantly, contain a small comment about what kind of ontology is represented as well as version control and used imports. These imports usually consist of used namespaces as well as other ontologies used. Version control may link to prior versions, version information, backwards compatibility or incompatibilities. The import is transitive. An example of this preamble can be seen in Figure 6.

```

1 <rdf:RDF
2   xmlns:owl = "http://www.w3.org/2002/07/owl#"
3   xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4   xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
5   xmlns:xsd = "http://www.w3.org/2001/XMLSchema#"
6
7 <owl:Ontology rdf:about="">
8   <rdfs:comment>An example OWL ontology</rdfs:comment>
9   <owl:priorVersion rdf:resource="http://www.mydomain.org/ahospital-old"/>
10  <owl:imports rdf:resource="http://www.mydomain.org/persons"/>
11  <rdfs:label>Animal hospital Ontology</rdfs:label>
12 </owl:Ontology>

```

Figure 6: Code snippet showing the first lines of a OWL1 ontology.

After this come the definitions making up the ontology. In OWL1 most classes, ontologies, and ontology elements are defined by IRIs. There are some exception to this, one example being anonymous individuals which do not need an IRI but still need a local identifier [51].

A class has two parts: a name or a reference and a list of restrictions. Which can be seen in Figure 7.

```

1 <owl:Class rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#
2   staffMember">
3   <rdfs:comment>The class of staff members</rdfs:comment>
4 </owl:Class>
5 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#veterinarian -->
6 <owl:Class rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#
7   veterinarian">
8   <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/delph/ontologies
9     /2023/0/petclinic#staffMember"/>
10  <rdfs:comment>The class of veterinarian All veterinarian are staff members.</
11  rdfs:comment>
12 </owl:Class>

```

Figure 7: RDFS/OWL code snippet for showcasing classes given in subsection 2.3.1 for an OWL2 ontology.

It is further possible to define instances belonging to different classes. A minimal instance is defined by declaring it to be a member of a class. An instance is only unique if explicitly qualified as such by the OWL assertion *owl:differentFrom* or similar assertions. This is a big difference when comparing OWL to a normal database language and can be seen in the OWL2 example seen in Figure 8. This example is used as for this assertion it pertains to both OWL1 and OWL2 and can as such be seen as an example for OWL1 here.

Properties exist to define facts about members or classes and individual instances. An OWL1 property is a binary relation and can be expressed by the formula  $P(a,b)$ . There are two types of properties. Datatype properties relating datatypes to instances of classes and object properties that relate two classes. These properties can be restricted in different ways, for example by specifying the domain as well as range. Properties can be arranged hierarchically like classes. Datatypes for datatype properties are either RDF literals or some of the XML schema datatypes [51]. Object properties can be symmetric or transitive. It is possible to define classes by boolean combinations like unions, intersections and complements of other classes [3]. There are two semantics available for OWL1 to infer meaning from a given ontology. One is based on RDF called *rdf-based semantics* and gives meaning to an ontology by looking at it as an RDF graph. Another is based on *description logic*



```

1 <rdf:Description>
2   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDifferent"/>
3   <owl:distinctMembers rdf:parseType="Collection">
4     <rdf:Description rdf:about="http://www.semanticweb.org/delph/ontologies
5 /2023/0/petclinic#Help_Pets_Inc"/>
6     <rdf:Description rdf:about="http://www.semanticweb.org/delph/ontologies
7 /2023/0/petclinic#Jane_Doe"/>
  </owl:distinctMembers>
</rdf:Description>

```

Figure 8: RDFS/OWL code snippet for showcasing unique individuals given in subsection 2.3.1 for an OWL2 ontology. The use of *owl:distinctMembers* makes the members in the collection unique.

called direct semantics. The direct semantics can be considered a subset of the rdf-based semantics [52].

When using OWL, it is of importance to keep in mind that OWL, like many ontology description languages, functions on the open-world-assumption. This can be seen by the fact that a different name does not mean two instances are unique, because no data exists verifying them as unique. As such, uniqueness must be explicitly specified as can be seen in the example from OWL2 Figure 8. However, as this is not always wanted there exists a subset of OWL1 that is not officially defined by the W3C, called OWL1 DLP, which is the largest subset of OWL1 for which the difference between the open-and closed-world-assumptions do not matter [4, 3].

As in OWL1, the primary syntax for OWL2 is the RDF/XML syntax, which is mandatory for all OWL2 tools to support. There are however, other syntaxes available and there are tools to translate between different syntaxes. As with OWL1, there are the same two semantics available: rdf-based semantics and direct semantics. The rdf-based semantic is completely compatible with OWL1 and the direct semantic is mostly compatible [50].

Like in OWL1, the full language is called OWL Full. There also exists a subset called OWL2 DL, which restricts the syntax of OWL Full for ease of implementation and computational decidability. These two versions of OWL are usually described using different semantics. In most cases OWL2 DL is expressed using direct semantic. An Ontology that goes beyond the OWL2 DL subset, being part of OWL2 Full, is usually using rdf-based semantics. In practice every ontology using direct semantic is seen as an OWL2 DL ontology. It is however possible to interpret OWL2 DL using rdf-based semantics and as such this isn't necessarily true. OWL2 Full using rdf-based semantic is undecidable and no reasoners exist, while OWL2 DL using direct semantic is decidable and reasoners exist [51, 50].

OWL2 DL is divided into sublanguages called profiles, by the W3C, which are split by computational efficiency. The sublanguages are listed in Table 2 and the relationship between them is visualized in Figure 9. Every subset ontology is of course a valid OWL2 ontology and is part of the OWL2 DL subset.

Table 2: Comparison of the OWL2 sublanguages.

OWL EL	OWL RL	OWL QL
Polynomial time reasoning (in respect to the ontology size)	Polynomial time reasoning with rule-extended database technologies.	Queries can be answered in LOGSPACE using relational database technology
Suitable for large ontologies	Suitable for lightweight ontologies with large number of individuals, where it is useful to operate on data via RDF triples	Suitable for for lightweight ontologies, with a large number of individuals, where query answering is important

A full list of the computational properties of every OWL2 sublanguage can be found at the W3C website<sup>1</sup>. It is also important to note that every profile is a structural restriction of OWL2 and as

<sup>1</sup>See: [https://www.w3.org/TR/owl2-profiles/#Computational\\_Properties](https://www.w3.org/TR/owl2-profiles/#Computational_Properties) Last checked 04.05.2023.

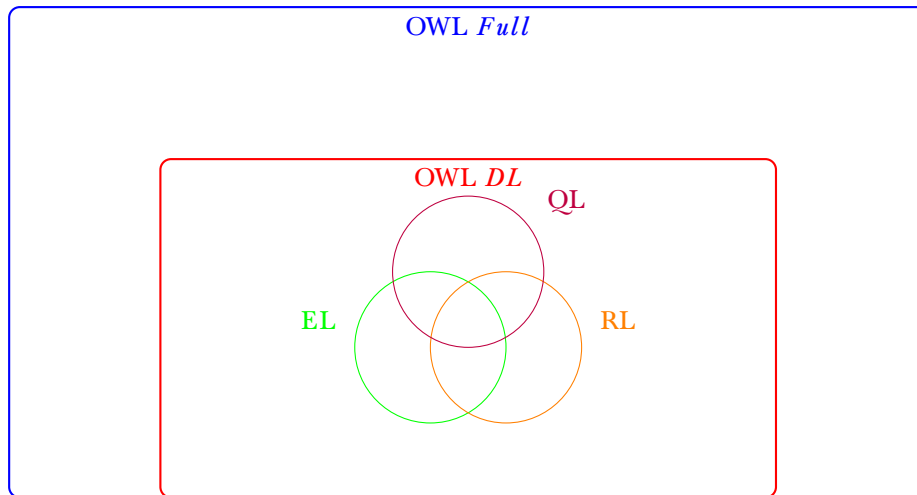


Figure 9: The relationship between the standard OWL2 sublanguages visualized.

such every OWL1 sublanguage can be viewed as an OWL2 profile. There also exist other profiles that are developed, but are not maintained standards by the W3C.

OWL2 is, as a whole, very similar to OWL1 and backwards compatibility is guaranteed. There are however, some changes between the two version. One of those changes is that IRI are used instead of URI to better support non-English languages. The use of IRIs also simplifies the versioning of OWL2, as there may be a version IRI to specify the ontology version [46].

In addition to the new sublanguages mentioned above, there are some other new features regarding syntax improvements, better datatype support, metamodeling capabilities, better annotation capabilities and expressivity by adding features for properties. A noteworthy feature is the ability to define keys for classes. Keys can be used to assign a set of key properties of a given class. These properties can be used to identify unique individuals. Furthermore, it is possible to assert object properties as reflexive, irreflexive or asymmetric. Meaning they respectively, are applicable to all individuals, are not applicable for any individual and a property holds between an individual  $x$  and  $y$  and not in the reverse [47, 48, 46]. The full sourcecode for the OWL example ontology in RDF/XML can be found in Appendix A. In the following chapters the OWL keyword will stand for OWL2 [3, pp. 113-135][49, 52, 51].

### 2.3.2. Existing Software





This section will give an overview over existing software for ontology development, testing, and maintenance. The focus will be on tools highlighted in the OWL primer of the W3C<sup>2</sup> as most prominent, as well as tools in use by the scientific community, as encountered in the research for this thesis. One example of this is found in Haghgoo et al. [22]. They introduce many of the featured tools, like Ontology Pitfall Scanner! (OOPS!) and Themis. This section highlights tools used for editing ontologies, tools used for logical inference and tools used for analysis. A subset of tools that can be used to search and compare ontologies is highlighted in section 5 and not in this chapter. Although there is overlap, explaining tools for expressively searching and comparing ontologies in section 5 is a better fit for reading comprehension.









**Ontology Editors** The first software that will be highlighted are so called ontology editors, which are used to create and edit ontologies. An overview can be seen in Table 3. The Ontology editor from Topbraid will not be looked at, as it is not open-source and the only free version available is no longer in development. Furthermore, the Neon toolkit will not be looked at as the last version is from 2011<sup>3</sup>. This is also the case for SWOOP, as its latest found release is from 2006 and the project website is defunct<sup>4</sup>.

<sup>2</sup>See: [https://www.w3.org/TR/owl2-primer/#OWL\\_Tools](https://www.w3.org/TR/owl2-primer/#OWL_Tools) Last checked 04.05.2023.

<sup>3</sup>[http://neon-toolkit.org/wiki/Main\\_Page.html](http://neon-toolkit.org/wiki/Main_Page.html) Last checked 04.05.2023.

<sup>4</sup><https://github.com/ronwalf/swoop> Last checked 04.05.2023.

Table 3: Relevant ontology editors. *MacOs*: , *Windows*: , *Linux*: , *Web-based*: .

Editor	Latest Release (year)	Syntaxes	Supported OS	Tutorial	Other
Protégé	5.5.0 (2019)	Not found on website Without Plugins: RDF/XML, Turtle, OWL/XML, OWL functional, Manchester OWL, OBO, Latex, JSON-LD	  	outdated	widely used, web version available plugins supported, wide variety of plugins, reasoner support
Eddy	3.3 (2022)	Graphol RDF/XML, Turtle, OWL/XML, Manchester OWL, OWL functional	  	Not on website, user manual in GitHub wiki	standard OWL2 profile support Build for Graphol ontology language, fully graphical based plugins supported hermit reasoner integrated syntax validation (to OWL2 profiles)
Vitro	1.13 (2022)	Notation3 RDF/XML, Turtle, N-Triples maybe more not clear		Not found	Backbone of vivo and other projects, fully web based used by different communitites world wide Fact++ reasoner integration
Vocbench3	11.3 (2022)	Not found		Yes on website	OWL2 support fully web based

Protégé<sup>5</sup> is the most used Ontology Editor in the semantic web community. It exists as a desktop tool as well as as a web program and has a wide selection of plugins maintained by the community. As it is one of the few editors that is easily usable, can be used on end user machines, has a wide variety of plugins and is still fairly up to date. Eddy<sup>6</sup> is a fairly new editor based on a graphical ontology language called Graphol. As it is new it remains to be seen if it will be widely adopted.

<sup>5</sup>See: <https://protege.stanford.edu/> Last checked 04.05.2023.

<sup>6</sup>See: <https://github.com/obdasystems/eddy> Last checked 04.05.2023.

Vitro<sup>7</sup> is not used on its own. It is a web based ontology editor that is part of many other projects to facilitate ontology editing. Vocbench<sup>8</sup> is the third iteration of a web based ontology editor that is funded by the European Union.

**Ontology Reasoners** The second suite of software used with ontologies that is highlighted are ontology reasoners. Ontology reasoners are used to test the integrity of ontologies as well as infer logical conclusion from the ontology. There exist many different reasoners written in different languages. Different reasoners support different profiles of OWL2, but almost all reasoners are only available for the OWL2 DL subset of OWL2. Reasoners are able to infer new facts from the given ontology. Furthermore, some reasoners can check for consistency of the ontology. There are commercial as well as open-source reasoners, this following section will briefly cover some open-source solutions. Protégé as well as Eddy come with in-built Hermit<sup>9</sup> reasoner support. Hermit is an OWL2 DL reasoner using direct-semantics built in java. Another popular reasoner is Fact++<sup>10</sup>, written in C++. It is available for Protégé by use of a plugin and comes per default with Vitro. Another quite popular reasoner, being Pellet<sup>11</sup>. Pellet is a Java based reasoner last updated 2016. An open-source fork exists called openllet<sup>12</sup>. The reasoners so far are based on direct semantics. There are however, some reasoners based on RDF-based semantics. One of which is the HyLAR-Reasoner<sup>13</sup>. HyLAR is a web based reasoner written in JavaScript. A list of all reasoners mentioned can be found in Table 4. There are some lists on the internet that try to compile reasoners, mainly a list from the W3C<sup>14</sup> and a list from the university of manchester<sup>15</sup> both are, however, not maintained and contain dead links.

Table 4: Some ontology reasoners.

Reasoner	Latest Release (year)	Programming Language	Semantics	Sublanguages
Hermit	1.3.8 (2013)	Java	direct	OWL2 DL
Fact++	1.6.5 (2016)	C++	direct	OWL2 DL
Pellet	2.3.1 (2016)	Java	direct	OWL2 DL
Openllet	2.6.5 (2019)	Java	direct	OWL2 DL
HyLAR-Reasoner	1.8.3 (2017)	JavaScript	rdf-based	OWL2 RL

**Ontology Analysis Software** The next software used with ontologies will be accumulated under the umbrella term analysis software in this work. This contains a variety of software that can be used to assist in analyzing ontologies. Reasoners with consistency checking capabilities, as well as ontology editors with support for analysis can be understood as falling within this category. They are listed separately, as this is not the main purpose of the reasoners and editors. This section will mainly focus on software that can be used in the comparison of ontologies in subsection 4.3. There are four types of software to be mentioned here. The first type of software can not be categorized as any of the following types. The second tool type being ontology visualizer, being able to visualize ontologies in a human readable graphical format. Ontology matching tools are the third type of software. They can be used to automatically map two ontologies. The fourth software tool are ontology editing libraries being used in different programming languages to work with ontologies. There can be some overlap between these categories.

One software tool for analyzing ontologies, that falls under the first category is OOPS!. This tool is used to find pitfalls, that can occur in the modeling of an ontology. The tool is able to semi-automatically detect 33 pitfalls from a list of pitfalls accumulated from different ontology evaluation works [54]. Further tools that work similar to OOPS! are the tools Ontology Pitfall Scanner for FAIR

<sup>7</sup>See: <https://github.com/vivo-project/Vitro> Last checked 04.05.2023.

<sup>8</sup>See: <http://vocbench.uniroma2.it/> Last checked 04.05.2023.

<sup>9</sup>See: <http://www.hermit-reasoner.com/> Last checked 04.05.2023.

<sup>10</sup>See: <https://bitbucket.org/dtsarkov/factplusplus/src/master/> Last checked 04.05.2023.

<sup>11</sup>See: <https://github.com/stardog-union/pellet> Last checked 04.05.2023.

<sup>12</sup>See: <https://github.com/Galigator/openllet> Last checked 04.05.2023.

<sup>13</sup>See: <https://github.com/ucbl/HyLAR-Reasoner> Last checked 04.05.2023.

<sup>14</sup>See: <https://www.w3.org/2001/sw/wiki/OWL/Implementations> Last checked 04.05.2023.

<sup>15</sup>See: <http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/> Last checked 04.05.2023.

(FOOPS!)<sup>16</sup> and Fair-Checker (FAIR-Checker)<sup>17</sup> [62][19]. Both tools aim to analyze the ontology according to the FAIR principles and are open source. The FOOPS! scanner gives a score according to each principle of FAIR as well as an overall score. The score is computed by 24 metrics that are split between the principles. The FAIR-Checker tool gives a score for eight metrics that are divided onto the four FAIR principles. Another tool, that does not fit in any of the other three categories, enabling the automatic contentual testing of ontologies is Themis<sup>18</sup>, which allows the user to test the logical structure of the ontology. A tool using Themis as well as OOPS! is OnToology<sup>19</sup>. This tool allows for the automatic production of documentation as well as testing of an ontology. It can be used by providing the GitHub repository of an existing ontology and a live version can be found here<sup>20</sup>.

There are many different libraries to visualize ontologies, two examples would be WebVOWL<sup>21</sup> and AR2DTool<sup>22</sup>. As this category of tools is not relevant for this thesis more will not be mentioned.

Another set of tools, which exist to match different ontologies with each other, are ontology matching tools. One example being LogMap<sup>23</sup>. LogMap can automatically match ontologies. This area of tools is in active development and the Ontology Alignment Evaluation Initiative held a campaign in 2022 comparing different mapping tools, the preliminary results of which can be found here<sup>24</sup>.

The fourth category of tools, ontology editing libraries, enable working with ontologies in different programming languages. Most of these libraries enable the loading, querying, creating and modifying of ontologies. There are some differences in the programming language used, speed and documentation. One example for Python being Owlready2<sup>25</sup>. It is well documented, allows for the modification, querying and creation of ontologies and also has in-built reasoner support by using Pellet or HermiT. Another library built on top of Owlready2 is ontor<sup>26</sup> which adds visualization support. A library for Java is the OWLAPI<sup>27</sup> a widely used Java library for crating and modyfing ontologies. Another OWL library for Rust is horned-owl<sup>28</sup>. Documentation is not easily found, however, it is faster in parsing ontologies than OWLAPI [36].

---

<sup>16</sup>See: [https://github.com/oeg-upm/fair\\_ontologies](https://github.com/oeg-upm/fair_ontologies) Last checked 04.05.2023.

<sup>17</sup>See: <https://github.com/IFB-ElixirFr/FAIR-checker> Last checked 04.05.2023.

<sup>18</sup>See: <https://github.com/oeg-upm/Themis> Last checked 04.05.2023.

<sup>19</sup>See: <https://github.com/OnToology/OnToology> Last checked 04.05.2023.

<sup>20</sup>See: <http://ontology.linkeddata.es/> Last checked 04.05.2023.

<sup>21</sup>See: <https://github.com/VisualDataWeb/WebVOWL> Last checked 04.05.2023.

<sup>22</sup>See: <https://github.com/idafensp/ar2dtool> Last checked 04.05.2023.

<sup>23</sup>See: <https://github.com/ernestojimenezruiz/logmap-matcher> Last checked 04.05.2023.

<sup>24</sup>See: <http://oei.ontologymatching.org/2022/results/> Last checked 04.05.2023.

<sup>25</sup>See: <https://owlready2.readthedocs.io/en/v0.37/> Last checked 04.05.2023.

<sup>26</sup>See: <https://github.com/felixocker/ontor> Last checked 04.05.2023.

<sup>27</sup>See: <https://github.com/owlcs/owlapiOne> Last checked 04.05.2023.

<sup>28</sup>See: <https://github.com/phillord/horned-owl> Last checked 04.05.2023.



### 3. Ontologies in Energy Research

This section focuses on the ontologies that are compared in this thesis. First, the criteria for choosing the ontologies are explained in subsection 3.1 and afterwards a short overview over the chosen ontologies is given in subsection 3.2.

#### 3.1. Method and Selection

It is not possible to compare every ontology in the energy sector in this thesis. As such, it is important to choose a suitable subset of ontologies to compare. This section aims to explain the selection process of this thesis as well as give an overview over the selected ontologies to answer the question which ontologies are used in energy research. To evaluate the landscape of ontologies that can be used for energy research, this thesis follows a systematic approach. The process for identifying ontologies is adapted from the PRISMA process. Figure 10 shows the process used with the number of ontologies/papers identified in each step.

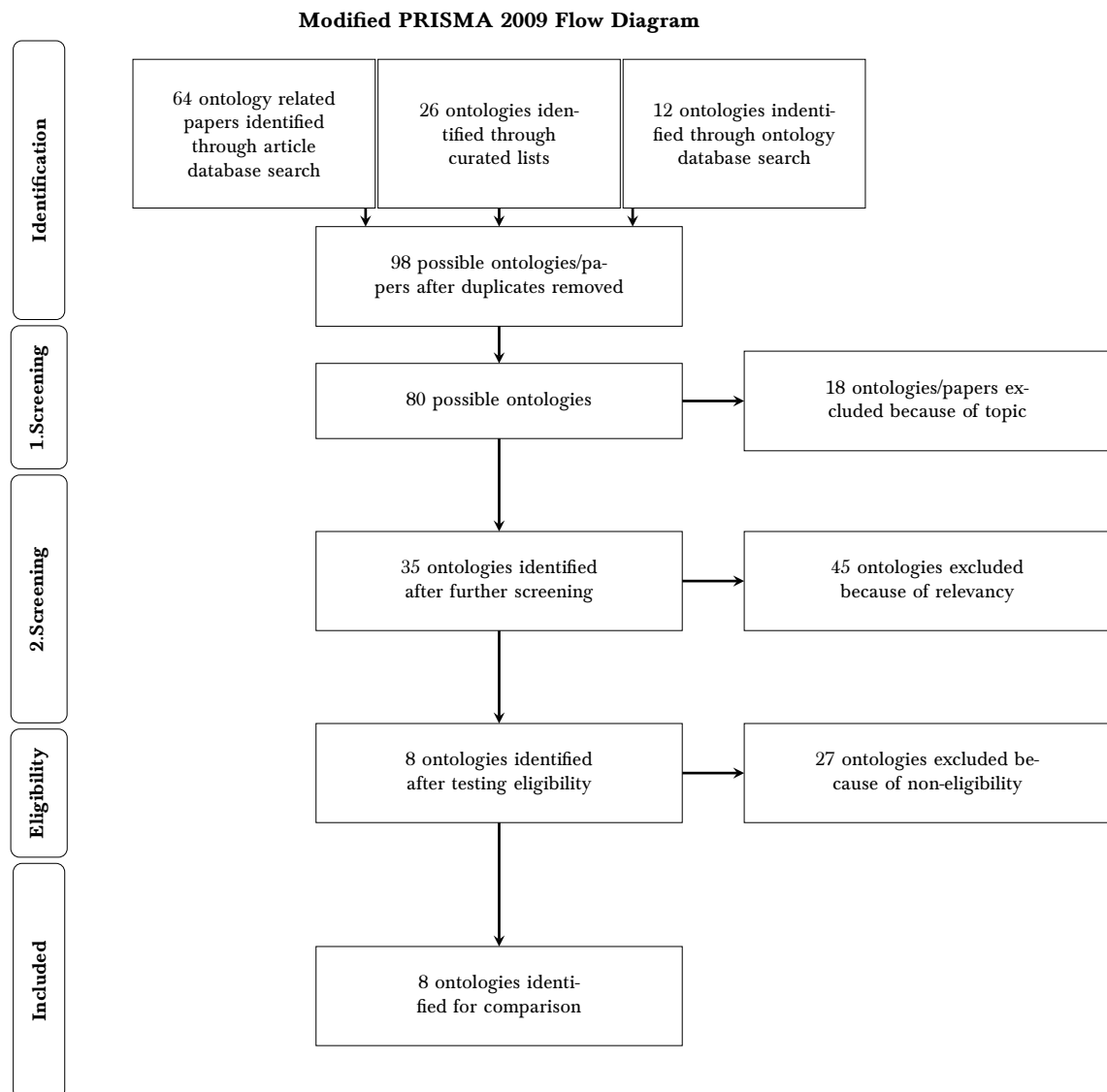


Figure 10: Modified Prisma approach used for ontology procurement

There are four phases to the ontology procurement, they are described in the following section:

**Identification:** For a preliminary identification, three sources are used. The first source is an Article Database search using Google Scholar<sup>29</sup>, Elsevier<sup>30</sup>, Institute of Electrical and Electronics Engineers (IEEE)<sup>31</sup> as well as Scopus<sup>32</sup> with the keywords energy and ontology. The second source is an ontology database search using the Linked Open Vocabularies (LOV) search engine<sup>33</sup> with the same keywords. The third source is the use of curated lists in other papers regarding the topic mainly Wierling et al. [71]. Through this search, 98 ontologies and papers about ontologies are identified after removing duplicates.

**1. Screening:** After the initial search, a screening is applied. Only ontologies that are in english are eligible. Furthermore, the ontologies need to have an energy topic as the focus or have a significant portion dedicated to an energy topic. In this phase of screening the papers that were identified in the article database search are filtered and papers that use existing, already known ontologies or do not fit the topic are removed, so that only papers about new ontologies remain. The screening is done by reviewing the title and abstract of given papers. Skimming through the papers to determine if they are about new ontologies if necessary. After this screening, 80 ontologies are identified and 18 papers are excluded, as they either use existing ontologies or do not fit the energy topic.

**2. Screening:** Following this initial screening, the ontologies are screened further with relevancy as the foremost criterion in mind. Since relevance is a vague term, it will be defined for this selection process. A relevant ontology is defined by two traits. The age of the ontology, a newer ontology being more relevant than an older ontology and the citations of an ontology, an often cited ontology being more relevant than a less cited ontology. Regarding the citation, the statistics given by Google Scholars<sup>34</sup> statistics are used. Furthermore, an ontology that is defined as a standard or funded through government backing is also categorized as more relevant. Using the method mentioned above, 35 ontologies were identified after the second screening. The full list of ontologies after this step can be found in Table 5. Out of the 35 ontologies, two are considered standards and 21 received government funding of some kind. Furthermore, 11 of the ontologies are ontologies about smart homes. 11 ontologies are generally about the energy domain. Eight are about the smart grid. Two are about energy management. Two are about certain renewable energies (wind and solar) and one is about the energy market

Table 5: Ontologies identified after second screening. Citations taken from Google Scholar. Citations last checked: 08.02.2023. Sorted by Citations descending.

Ontology	Topic	Year	Reference	Standard	Citations	Gov. Funding
DNAs ontology	Smart home	2019	[24] [25]	✗	355,172	✗
Brick	Smart home	2016	[5]	✗	264	✓
ThinkHome	Smart home	2011	[60]	✗	219	✓
A comprehensive smart home ontology	Smart home	2012	[29]	✗	199	✓
BOnSAI	Smart home	2012	[67]	✗	147	✓
A performance assessment ontology	Smart home	2015	[10]	✗	114	✓
SEAS Ontology	Energy domain	2017	[34]	✗	78	✓

Continued on next page

<sup>29</sup>See: <https://scholar.google.de/> Last checked 04.05.2023.

<sup>30</sup>See: <https://www.sciencedirect.com/> Last checked 04.05.2023.

<sup>31</sup>See: <https://ieeexplore.ieee.org/Xplore/home.jsp> Last checked 04.05.2023.

<sup>32</sup>See: <https://www.scopus.com> Last checked 04.05.2023.

<sup>33</sup>See: <https://lov.linkeddata.es/dataset/lov/> Last checked 04.05.2023.

<sup>34</sup>See: <https://scholar.google.de/> Last checked 04.05.2023.



Table 5: Ontologies identified after second screening. Citations taken from Google Scholar. Citations last checked: 08.02.2023. Sorted by Citations descending. (Continued)

Ontology	Topic	Year	Reference	Standard	Citations	Gov. Funding
SAREF4ENER	Smart home	2016	[14]	✓	59	✗
Facility ontology	Energy management	2015	[69]	✗	55	✗
Energy-saving ontology	Energy domain	2013	[73]	✗	52	✗
DEHEMS	Energy Management	2011	[65]	✗	47	✓
ProSGV3	Smart Grid	2014	[20]	✗	41	✗
OEMA	Energy domain	2017	[10]	✗	41	✗
OntoPowSys	Smart grid	2020	[16]	✗	36	✓
PQONT	Energy domain	2010	[33]	✗	33	✗
EM-KPI	Smart Grid	2019	[35]	✗	32	✓
OEO	Energy domain	2021	[7]	✗	30	✓
DIMMER ontology	Smart grid	2016	[8]	✗	28	✓
Hybrid energy systems ontology	Smart grid	2017	[63]	✗	27	✗
Building Energy Ontology	Smart home	2019	[37]	✗	24	✓
EE-district ontology	Smart grid	2016	[8]	✗	23	✓
Electricity Markets Ontology	Energy Markets	2016	[64]	✗	22	✓
SARGON	Energy domain	2020	[22]	✗	17	✓
Solar soft cost ontology	Solar Energy	2019	[6]	✗	16	✓
DABGEO	Energy domain	2020	[12]	✗	14	✓
Generic Ontology of Energy Consumption Households	Smart home	2019	[30]	✗	12	✓
Onto-SB	Smart home	2018	[15]	✗	12	✗
OntoWind	Wind Energy	2018	[32]	✗	12	✗
NewOSEIM	Smart home	2021	[10]	✗	11	✗
OpenADR Ontology	Smart Grid	2020	[44]	✗	10	✓
EnArgus ontology	Energy domain	2021	[45]	✗	7	✓
DECENT	Energy domain	2021	[28]	✗	6	✗
CIM	Energy domain	2003	[26]	✓	3	✓

Continued on next page

Table 5: Ontologies identified after second screening. Citations taken from Google Scholar. Citations last checked: 08.02.2023. Sorted by Citations descending. (Continued)

Ontology	Topic	Year	Reference	Standard	Citations	Gov. Funding
∇ Platform ontology	Energy domain	2022	[41]	✗	0	✗
Integrated heat and electric energy ontology	Smart grid	2023	[31]	✗	0	✗

**Eligibility:** After identifying relevant ontologies via two screenings, the final selection is made. This is necessary, as the goal of this thesis is an in-depth comparison. In this context such a comparison only possible with a selection of five to ten ontologies because of time constraints. The final selection is furthermore made with the research question posed in section 1 in mind. As such, the focus lies on ontologies that cover a wide range of energy research topics. A further selection criterion in this part of the process is the restriction to at most one ontology per energy research topic, e.g. only a maximum of one ontologies with the focus on smart homes. For general energy ontologies, no such restriction is set. In this step care is also taken regarding the availability of the ontologies, as this is important for comparison. After this eligibility selection, eight ontologies are chosen, as can be seen in Table 6:

Table 6: Ontologies identified for comparison. Citations taken from Google Scholar. Citations last checked: 08.02.2023. Sorted alphanumerical by ontology name.

Ontology	Topic	Year	Reference	Standard	Citations	Gov. Funding
Brick	Smart home	2016	[5]	✗	264	✓
CIM	Energy domain	2003	[26]	✓	3	✓
DABGEO	Energy domain	2020	[12]	✗	14	✓
EMO	Energy Markets	2016	[64]	✗	22	✓
EM-KPI	Smart Grid	2019	[35]	✗	32	✓
OEO	Energy domain	2021	[7]	✗	30	✓
SARGON	Energy domain	2020	[22]	✗	17	✓
SEAS	Energy domain	2017	[34]	✗	78	✓

Of the ontologies chosen five are general ontologies covering the energy domain. One of them has a focus on the smart home and, one on the smart grid and another on energy markets.

### 3.2. Selected ontologies

This section gives a short overview of the chosen ontologies.

**Brick Ontology (Brick)** Brick is an ontology developed for smart homes. It was introduced in 2016 by Balaji et al. [5]. The ontology is cited 264 times according to Google Scholar. It is build to model sensors and subsystems in a building as well as the relationship between them.

It is chosen as it is widely cited and relevant ontology in the smart home domain. There exist a wide number of ontologies in this domain and it is a good representation for the domain.

**Common Information Model (CIM)** CIM is an ontology developed as an industry standard model for enabling information exchange between entities. It was introduced in 2003 by the IEC [26]. The ontology is cited three times according to Google Scholar. The core CIM standard enables the user to model every object needed in a normal electricity grid.

It is chosen as it is a standard that is widely used in Europe. There exist many different extensions for it and it can be used in many different applications.

**Domain Analysis-Based Global Energy Ontology (DABGEO)** DABGEO is an ontology network developed as a general energy domain ontology. It was introduced in 2020 by Cuenca, Larrinaga, and Curry [12]. The ontology is cited 14 times according to Google Scholar. It was developed after the OEMA ontology, also developed by Cuenca, Larrinaga, and Curry [11], as a successor ontology. In DABGEO the authors follow a similar structure to OEMA. The ontology consists of many smaller domains, each with their own ontology. The top-level domains covered by DABGEO are energy equipment, infrastructure, energy performance data, energy external factors and smart grid stakeholders.

It is chosen as it is build as an ontology network. The structure of the ontology makes it interesting for the comparison. DABGEO is looked at instead of the OEMA ontology since it provides better reusability and usability balance according to the authors. It is also a newer ontology.

**Electricity Markets Ontology (EMO)** EMO is an ontology developed to be used for modeling energy markets. It was introduced in 2016 by Santos et al. [64]. The ontology is cited 22 times according to Google Scholar. It was developed to support the agent based simulation of electricity markets. It is an ontology used in six different ontologies. Three electricity markets ontologies, a call for proposal ontology, a results ontology and a interoperability ontology make up these six different ontologies.

It is chosen as it is an ontology in a domain where not many ontologies exist. Furthermore, the approach to look at the energy domain from the market side is represented through this ontology.

**Energy Management Key Performance Indicator Ontology (EM-KPI)** EM-KPI is an ontology developed to exchange performance data between buildings to facilitate energy management. It was introduced in 2019 by Li et al. [35]. The ontology is cited 32 times according to Google Scholar. It is built reusing existing ontologies and introducing new elements. It consists of seven modules: the key performance index module, the observation model, the location module, the infrastructure module, the occupancy module, the weather module and the energy parameter module.

It is chosen as it is an ontology to represent the smart grid sector.

**Open Energy Ontology (OEO)** OEO is an ontology of the energy domain. It was introduced in 2021 by Booshehri et al. [7]. The ontology is cited 30 times according to Google Scholar. The researchers developed this ontology based on the Basic Formal Ontology (BFO) which is an upper ontology. The ontology is built around the Open Energy Platform (OEP) with a focus on open access and community collaboration, as it is freely available on GitHub to work on. It is developed with a focus on modeling energy systems as well as analyze them. OEO consists of four modules: oeo-model, a module concerning entities for models, data and more, oeo-physical a module covering physically existing entities, oeo-social covering social relations and people and oeo-shared covering concepts that are needed in all other modules [7].

It is chosen as it is an ontology that was developed recently and has a focus on community collaboration.

**Smart Energy Domain Ontology (SARGON)** SARGON is an ontology of the energy domain. It was introduced in 2020 by Haghgoo et al. [22]. The ontology is cited 17 times according to Google Scholar. It consists of 5 modules and is built as an extension of Smart Application Reference Ontology (SAREF). The ontology is developed to model the smart energy domain and is not an official extension by the European Telecommunications Standards Institute (ETSI). It should be mentioned

though, since SAREF is an Internet of Things (IoT) ontology after all and as such a certain focus on Internet of Things (IoT) exists in SARGON.

It is chosen as it is an ontology that is based on SAREF which is an official ETSI standard, but has a wider focus. Furthermore, a comparison with thee similar SEAS ontology seems interesting.

**Smart Energy Aware Systems Ontology (SEAS)** SEAS is an ontology of the energy domain. It was introduced in 2017 by Lefrançois [34]. The ontology is cited 78 times according to Google Scholar. It consists of four core modules and has several extensions which the authors call vertical modules that focus more on a domain inside of the energy domain. It also has a certain focus on the Internet of Things (IoT) domain, as it is based on SAREF which is an Internet of Things (IoT) ontology.

It is chosen as it is an ontology that is based on SAREF which is an official ETSI standard, but has a wider focus. Furthermore, a comparison with thee similar SARGON ontology seems interesting.

## 4. Analysis of Ontologies

This section analyzes the ontologies given in subsection 3.2 to find differences between them and determine a good general ontology for energy research. To achieve this, the criteria used in the analysis are given, explained and afterwards used to analyze and compare the ontologies. Furthermore, a score is calculated for easier comparison.

### 4.1. Computing a Score

To answer the question: “*Which ontology is suitable for a project in energy research?*”, postulated in subsection 1.1, a score for each criteria category is computed that helps identify the subjectively best ontology in each category. It is emphasized that, although it is composed to be as objective as possible, the computed score is a subjective value assigned to each ontology and as such might not fit the comparison research for every researcher. A score is still computed, because it allows for faster comparison and researchers that might not agree with the score are still able to evaluate the criteria on their own without relying on a score.

The score is computed by giving each criterion used in the score a value between zero and three, with zero being the worst and three being the best value that can be achieved in the given criterion. A criterion is used in the score if it can be quantified and is important for the evaluation of its category. The score for a given category is the mean of the individual values given rounded to the first decimal place. The following formula can be given for the score:

$$F = \frac{1}{n} \cdot \sum_{i=1}^n a_i,$$

where  $a_i \in [0,3]$  is the score for given for a criteria,  $n$  is the number of quantified criteria, and  $F \in [0,3]$ , is the score per category.

### 4.2. Criteria for Analysis of Ontologies

This section takes a look at existing comparisons in research, lists the criteria used in this comparison and explains them in detail.

#### 4.2.1. Related Work

This section showcases related work done in research to answer the question how ontologies can be compared.

Most works that compare ontologies do so in the context of a review of a sector, e.g. Grubic and Fan [21] who compare supply chain ontologies. These reviews/analyses of ontologies focus on comparing the content and scope of an ontology as well as other factors in that general direction and compare them based on a specific area of application in mind. Another paper by Lourdasamy and John [38] shows metrics for ontology evaluation which are mostly based on the ontology content. In this work the number of classes and other such attributes are taken into account and analyzed. Other than that some papers that introduce an ontology compare ontologies that are similar in nature. The comparison in papers like this is done as an aside and is mostly comparing content/showcasing which content is missing in other ontologies. Most comparisons are done in tables that showcase the different comparison criteria against the ontologies as well as textual, some are done completely textual.

In the energy sector ontologies are compared in the same way. There are some papers in which a general sector of ontologies is reviewed. Such as, Lygerakis, Kampelis, and Kolokotsa [39] who compare ontologies and other methods for energy efficiency in buildings. Other such papers by Wierling et al. [71] and Pritoni et al. [55] are also review energy related ontologies. Most papers that introduce a new energy ontology compare similar ontologies. As mentioned however, these comparisons are not the focus and focus on what is missing from other ontologies content-wise in comparison. Some examples for this include Haghgoo et al. [22] and Cuenca, Larrinaga, and Curry [12]. Both of which showcase different ontologies in the same sector and outline which content is missing.

For this comparison, the focus is not only on the content of the ontology, but rather on the metadata that exists around the ontology. The relevant criteria for this comparison can be found in the following

section.

#### 4.2.2. Criteria Categories

The criteria are divided into four categories according to the general topic they help analyze. This division is to structure the comparison and make it more digestible. The categories are in no way strict and should not be taken as objective division. The categories are:

Table 7: Ontology comparison categories.

Category	Description
Best Practice	Criteria that are associated with best practices to be followed in ontology creation.
Practical Implementation	Criteria that concern metadata that arises during the implementation of the ontology.
Maintenance and Accessibility	Criteria that concern the maintenance of the released ontology.
Governance	Criteria that concern the governance of the ontology.

The criterion in these categories are devised partly from ontology creation guides such as Arp, Smith, and Spear [4] or Antoniou and Van Harmelen [3] and partly devised from looking at the available ontology metadata and deciding what is useful in choosing an ontology. The following section explains the criteria sorted according to the different categories they belong to. It also defines how the criteria are listed in the tables in subsection 4.3. For the criterion that are used in calculating the score (see subsection 4.1), the score for each possibility is also given. If there is no score given the criterion is not used in the score.

#### 4.2.3. Best Practice

**Used Upper Ontologies** This criterion lists the Upper Ontologies the ontology is based on. It only lists the upper ontologies if the given ontology is based on it, not if only a part of it is used.

*Not Found:* No Upper Ontology used. (Score: 0)

*Upper Ontology Used:* Upper Ontology used. (Score: 3)

**Scope** This criterion lists the planned scope of the ontology planned by the authors. This criterion is qualitative by nature and consists of a short explanation of the ontology theme.

**Creation Type** This criterion lists the method used to create the ontology. The following creation methods are identified:

*Automatic:* Ontology created automatically by algorithm or other automated measure. Without or with little human interference. (Score: 0)

*Manual:* Ontology created manually by a human. (Score: 3)

*Hybrid:* Ontology created semi-automatically by using algorithms or automated measures and humans. (Score: 3)

If the creation method used is a well documented method, it is listed as such.

**Modularity** This criterion quantifies the modularity of the compared ontology. As ontologies are modular by design this criterion shows how easily modular a given ontology is. It is divided into two cases:

*Not (as) Modular:* Ontology does not consists of easily differentiable modules. (Score: 0)

*Modular:* Ontology consists of different modules that can easily be used independently. (Score: 3)

**Extensibility** This criterion quantifies how extensible a given ontology is. It is in part based on how modular an ontology is and also based on how well it is explained, e.g. explanations of the structure in additional documents. This criterion is divided into the following cases:

*Not (as) Extensible:* Ontology is not that easily extensible. (Score: 0)

*Explanations given:* Ontology has explanations that help newcomers to extend it by explaining it/the method used to create it further. (Score: 2)

*Modular:* Ontology is modular and through this more extendable. (Score: 2)

*Extensible:* Ontology is modular and has explanations that help newcomers to extend it by explaining it/the method used to create it further. (Score: 3)

**Validation** This criterion explains the steps the ontology creators used to validate the ontology. It is divided into the following cases:

*Not Validated* No Validation method given. (Score: 0)

*Validated through other means:* Some other means were used in the validation of the ontology. (Score: 1)

*Validated through automated Tools:* Automated tools, e.g. OOPS!, were used in validation (Score: 2)

*Validated through Experts:* Ontology was validated by experts in the field either using case studies, coverage studies or some other way. (Score: 2)

*Validated:* Automated tools and experts were consulted in the validation (Score: 3)

#### 4.2.4. Practical Implementation

**Available Languages** This criterion lists the different natural languages the ontology is available in.

**Available Ontology Languages** This criterion lists the different ontology languages the ontology is available in. It will be listed as a number in the comparison. It is divided into three different categories:

*Other:* Ontology is available in other languages. (Score: 0)

*RDF/XML:* Ontology is available in RDF/XML. (Score: 2)

*OWL:* Ontology is not available in RDF/XML, but in other OWL ontology languages. (Score: 2)

*RDF/XML+:* Ontology is available in RDF/XML and other ontology languages. (Score: 3)

**Description Size** This criterion looks at a random subset of 5 terms to determine the size of the descriptions. The terms are compiled using a Python script as outlined in subsection 4.2.7. If there are classes with no descriptions given in the selection, the rest of the terms are analyzed, but one point less is given for the score with a lower limit of zero. It is divided into three possible categories based on what the majority of the five terms descriptions are:

*Large* Over five sentences of description. (Score: 0)

*Medium:* Up to five sentences of description. (Score: 2)

*Small:* Up to two sentences of description. (Score: 3)

**Description Quality** This criterion looks at the same random subset of 5 terms used to determine the description size to determine the quality of the descriptions. The terms are compiled using a Python script as outlined in subsection 4.2.7. If there are classes with no descriptions given in the selection, the rest of the terms are analyzed, but one point less is given for the score with a lower limit of zero. It is divided into three possible categories based on what the majority of the five terms descriptions are and is more subjective:

*Hard:* Only understandable by domain experts. (Score: 0)

*Neutral:* Understandable by people with a slight understanding of the energy domain. (Score: 2)

*Easy:* Easily understandable even by people that are not experts in the energy domain. (Score: 3)

**Number of Terms** This criterion lists the dimension of the ontology. In this thesis the dimension means the number of classes the ontology envelops as well as the number of properties. The data is compiled using a Python script as outlined in subsection 4.2.7.

**Used Ontologies** This criterion lists all ontologies used in the ontology being compared. This criterion differs from the Used Upper Ontologies criterion layed out above as it lists all ontologies used and not only the upper ontology the compared ontology is based on. A used ontology is identified as given an ontology that is used with a namespace in the compared ontology. The data is compiled using a Python script as outlined in subsection 4.2.7.

#### 4.2.5. Maintenance and Accessibility

**Sourcecode** This criterion lists how many sources the ontology has. The ontology is looked for in different ontology database searches (LOV<sup>35</sup>, Leibniz-Informationszentrum Technik und Naturwissenschaften und Universitätsbibliothek (TIB) Terminology Service<sup>36</sup>, Ontology Lookup Service (OLS)<sup>37</sup>). Furthermore, it is looked for on GitHub<sup>38</sup> as well as sources listed in the paper presenting the ontology, if it is available. This criterion is divided into the following categories:

*0*: Ontology has no available sources. (Score: 0)

*1S*: Ontology has one available source, the source is self hosted. (Score: 2)

*1D*: Ontology has one available source, the source is not hosted by the authors. (Score: 2)

*2+*: Ontology has more than one available source. (Score: 3)

**License** This criterion defines the license of the ontology it is divided into two categories:

*Not Found*: Ontology license not found. (Score: 0)

*License Found*: Ontology license found. (Score: 3)

**Accessibility** This criterion defines the accessibility of a given ontology. It is divided into three categories:

*Closed-Access*: Ontology is not openly accessible. (Score: 0)

*Open-Access*: Ontology is openly accessible. (Score: 1)

*Open-Source*: Ontology sourcecode is openly available and openly accessible. (Score: 2)

*Open-Source (OSI)*: Ontology is open-source with a following open source initiative (OSI)<sup>39</sup> and openly accessible. (Score: 3)

**Maintenance** This criterion lists the persons or institution maintaining the ontology. Furthermore, it lists if the ontology is actively maintained. An actively maintained ontology is identified as either discourse taking place on GitHub in the issues tab or on the ontology website or the ontology was updated/revised in 2022. The criterion is divided into two categories:

*No Maintenance found*: No Maintenance found. (Score: 0)

*Active Maintenance*: Maintenance found. (Score: 3)

**Latest Release** This criterion lists the date of the latest release as well as the version. It is useful to determine how current the ontology is.

**Automatic Analysis** This criterion uses the tools OOPS!<sup>40</sup>, FOOPS!<sup>41</sup> and FAIR-Checker<sup>42</sup> to automatically analyze the ontology. A mean of the FAIR-Checker score is computed for each element (F,A,I,R). Furthermore, the Hermit reasoner is run to check for inconsistencies. The data is compiled using a Python script as outlined in subsection 4.2.7. If the ontology can not be analyzed as one, all parts of the ontology are analyzed and a mean is calculated. The criterion is divided into three categories:

*Problems*: Problems detected. (Score: 0)

*Minor Problems*: Less than two minor problems detected, e.g. minor pitfall according to OOPS!.

Overall FOOPS! score over 50 percent and less than two scores of zero in the mean FAIR-Checker score. (Score: 1)

*No Problems*: No problems detected in the analysis. (Score: 3)

<sup>35</sup>See: <https://lov.linkeddata.es/dataset/lov/> Last checked 04.05.2023.

<sup>36</sup>See: <https://service.tib.eu/ts4tib/index> Last checked 04.05.2023.

<sup>37</sup>See: <https://www.ebi.ac.uk/ols/index> Last checked 04.05.2023.

<sup>38</sup>See: <https://github.com/> Last checked 04.05.2023.

<sup>39</sup>See: <https://opensource.org/licenses/> Last checked 04.05.2023.

<sup>40</sup>See: <https://oops.linkeddata.es/> Last checked 04.05.2023.

<sup>41</sup>See: [https://foops.linkeddata.es/FAIR\\_validator.html](https://foops.linkeddata.es/FAIR_validator.html) Last checked 04.05.2023.

<sup>42</sup>See: <https://fair-checker.france-bioinformatique.fr/check> Last checked 04.05.2023.



#### 4.2.6. Governance

**Funding** This Criterion outlines where the funding that was used in the development of the ontology came from. It is differentiated between the number of source of funding.

*No Funding found:* No funding found. (Score: 0)

*Single Source of Funding Found:* Single source of funding found. (Score: 1)

*Multiple Sources of Funding Found:* Multiple sources of funding found. (Score: 3)

**Governing Instances** This criterion lists the governing Instances for the ontology. A governing instance being a person or institution with the final say on the content of the ontology.

*No Governing Instances found:* No Governing Instances found. (Score: 0)

*Governing Instances:* Governing Instances found. (Score: 3)

**Citations** This criterion lists the citation of the paper introducing the ontology as determined via Google Scholar<sup>43</sup>.

#### 4.2.7. Python Script for Data Compilation

A Python script is used to aid in the data compilation of some of the criteria. The script uses the package Owlready2<sup>44</sup> to load ontologies and choose five classes for the description quality and description size criteria with the help of the random module. If the ontology can not be loaded with Owlready2, the classes are picked manually. Furthermore, Owlready2 also used to compile the number of classes and properties for the number of terms criterion if the ontology can be loaded.

The library rdflib<sup>45</sup> is used to compile the ontologies for the used ontologies criterion if possible.

Furthermore, the request package and rdflib library are used to automatically generate OOPS<sup>46</sup>, FOOPS!<sup>47</sup> and FAIR-Checker<sup>48</sup> report for the automatic analysis criterion. The script also generates a mean FAIR score from the FAIR-Checker report as described in paragraph 4.2.5.

Furthermore, the script is used in case of the DABGEO ontology network to sum the classes and used ontologies as well as mean the reports for the automatic analysis. As it is an ontology consisting of many smaller ontologies and it is not possible to create only one report.

The Python script can be found on GitHub<sup>49</sup> under the MIT license.

### 4.3. Analysis of the Ontologies

The ontologies that were chosen in subsection 3.2 are compared in this section. The comparison is divided into the categories outlined in Table 7. The criterion results are aggregated in truncated form in four tables. The tables also show the calculated score and are colorcoded to mark positive (green), neutral (yellow) and negative (red) traits. The accumulated data can be found in unordered form on zenodo<sup>50</sup> and a structured form of the comparison can be found on the Open Research Knowledge Graph (ORKG)<sup>51</sup>.

#### 4.3.1. Issues during Data Collection

During the data collection, certain issues came up, which are explained in more detail in this section. It should be mentioned that in OEO the class descriptions are not found as *rdfs:comment* tags but as *obo:IAO\_0000115*. Furthermore, the automated script was not able to load Brick, SEAS and 40 of the 97 ontologies that are part of DABGEO. Because of that the classes used for identifying description size and description quality were chosen by hand. For Brick and SEAS the number of terms was identified through Protégé. For DABGEO the numbers given in Cuenca, Larrinaga, and Curry [12] introduction paper were used. For the automatic analysis of DABGEO the individual ontologies were

<sup>43</sup>See: <https://scholar.google.de/> Last checked 04.05.2023.

<sup>44</sup>See: <https://github.com/pwin/owlready2> Last checked 04.05.2023.

<sup>45</sup>See: <https://github.com/RDFLib/rdflib> Last checked 04.05.2023.

<sup>46</sup>See: <https://oops.linkeddata.es/> Last checked 04.05.2023.

<sup>47</sup>See: [https://foops.linkeddata.es/FAIR\\_validator.html](https://foops.linkeddata.es/FAIR_validator.html) Last checked 04.05.2023.

<sup>48</sup>See: <https://fair-checker.france-bioinformatique.fr/check> Last checked 04.05.2023.

<sup>49</sup>See: <https://github.com/SupaaSchnitzel/OntMetaScript>. The final version for this thesis is found under the tag *v1.0*. Last checked 20.05.2023.

<sup>50</sup>See: <https://doi.org/10.5281/zenodo.7895261> Last checked 17.05.2023.

<sup>51</sup>See: <https://orkg.org/comparison/R592041/> Last checked 17.05.2023.

analyzed and a mean score was generated for the FOOPS! and Fair score. For 11 of the ontologies, however, no OOPS! score was generated, as the REST service of Poveda-Villalón, Gómez-Pérez, and Suárez-Figueroa [54]<sup>52</sup> was not able to generate a report. Furthermore, some of the links on the DABGEO website<sup>53</sup> are not correctly configured, e.g. the link for the air pollutants ontology<sup>54</sup> on the hyperlink for the ontology on the landing page is wrong and should be a different link<sup>55</sup>.

The website of the EMO ontology is not reachable via https and EMO is not automatically downloaded via the link provided. The sourcecode either needs to be copied into a file or downloaded via wget<sup>56</sup> or other such means.

#### 4.3.2. Best practice

In this category the ontologies are compared according to their conformance to the best practices in ontology design. Table 8 gives an overview of the different criteria in the category.

Table 8: A comparison of the adherence to best practices for energy research ontologies. *Not found:*  $\times$ .

Metric \ Ontology	Best Practice						
	Used Upper Ontology	Scope	Creation Type	Modularity	Extensibility	Validation	Score
Brick	$\times$	Buildings	Manual Own Method	Modular	Extensible	Validated through Experts	2.2
CIM	$\times$	General	Manual Own Method	Modular	Modular	Validated through Experts	2.0
DAB-GEO	$\times$	General	Manual OntoCape Method [40]	Modular	Extensible	Validated through Experts	2.2
EMO	$\times$	General	Manual Own Method	Not as Modular	Explanations given	Not Validated	1.0
EM-KPI	$\times$	Electricity Market	Manual NeOn Method [68]	Not as Modular	Extensible	Validated through Experts	1.6
OEO	BFO	General	Manual OBO Foundry Method [1]	Modular	Extensible	Validated through Experts	2.8

Continued on next page

<sup>52</sup>See: <https://oops.linkeddata.es/webservice.html> Last checked 04.05.2023.

<sup>53</sup>See: <https://innoweb.mondragon.edu/ontologies/dabgeo/> Last checked 04.05.2023.

<sup>54</sup>See: [https://innoweb.mondragon.edu/ontologies/dabgeo/domain-task/application\\_type/city\\_energy\\_performance\\_assessment/airpollutants/1.0/ontology/](https://innoweb.mondragon.edu/ontologies/dabgeo/domain-task/application_type/city_energy_performance_assessment/airpollutants/1.0/ontology/) Last checked 04.05.2023.

<sup>55</sup>See: [https://innoweb.mondragon.edu/ontologies/dabgeo/domain-task/application\\_type/city\\_energy\\_performance\\_assessment/airpollutants/1.0/](https://innoweb.mondragon.edu/ontologies/dabgeo/domain-task/application_type/city_energy_performance_assessment/airpollutants/1.0/) Last checked 04.05.2023.

<sup>56</sup>See: <https://www.gnu.org/software/wget/> Last checked 04.05.2023.

Table 8: A comparison of the adherence to best practices for energy research ontologies. *Not found: X*. (Continued)

Metric \ Ontology	Best Practice						
	Used Upper Ontology	Scope	Creation Type	Modularity	Extensibility	Validation	Score
SAR-GON	X	General	Manual Linked open terms Method [53]	Modular	Modular	Validated	2.2
SEAS	X	General	Manual Own Method	Modular	Extensible	Not Validated	1.8

As can be seen in the above table, all ontologies with the exception of OEO do not use an upper ontology for their base.

While all of the listed ontologies were created manually, only half of them were created following a specific creation method that was established in research and the other half was created using methods that were not previously established in literature. Of the ontologies that were created using an established method, none of them used the same method.

Only SARGON uses automated validation tools as well as a reasoner for validation. Furthermore, the way researchers evaluate ontologies differ greatly. Balaji et al. [5] and Li et al. [35] (Brick, EM-KPI) employ use cases/examples to validate their ontologies. While Booshehri et al. [7] (OEO) employ a coverage study, competency questions, and an agreement study. The researchers developing SARGON employ use cases for validation as well as automated validation through OnToology. DABGEO is validated concerning reusability and usability as Cuenca, Larrinaga, and Curry [12] set this as their main objective. CIM is validated through the International Electrotechnical Commission (IEC).

#### 4.3.3. Practical Implementation

This category compares the practical implementation of the ontologies. Table 9 gives an overview over the criteria.

Table 9: A Comparison of the practical implementation of energy research ontologies.

Metric \ Ontology	Practical Implementation						
	Available Languages	Available Ontology Languages	Descr. Size	Descr. Quality	Number of Terms	Used Ontologies	Score
Brick	en	OWL Formats: ttl	Small*	Easy*	1623	11	2.0
CIM	en, fr <sup>†</sup>	RDF/XML+ Formats: owl,ttl	Small	Hard	10922	5	2.0
DAB-GEO	en	RDF/XML+ Formats: owl,ttl,n3	Small*	Easy*	2523	68	2.3

Continued on next page

Table 9: A Comparison of the practical implementation of energy research ontologies. (Continued)

Metric \ Ontology	Practical Implementation						
	Available Languages	Available Ontology Languages	Descr. Size	Descr. Quality	Number of Terms	Used Ontologies	Score
EMO	en	RDF/XML Formats: owl	Small	Easy	52	5	2.7
EM-KPI	en	RDF/XML+ Formats: owl,ttl,n3	Small	Easy	281	7	3.0
OEO	en	RDF/XML+ Formats: owl,omn	Small	Easy	1635	12	3.0
SAR-GON	en	RDF/XML+ Formats: owl,ttl	Small*	Easy*	266	6	2.3
SEAS	en	RDF/XML+ Formats: owl,ttl,n3,vowl	Small	Easy	901	9	3.0

\* The picked classes include some with no description given

† Only an english version of CIM is available as sourcecode.

All ontologies only have sourcecode with english language descriptions available. As can be seen in Table 9, most ontologies follow similar implementation practices. Except for Brick and EMO all of the ontologies have more than one form of ontology language available. Except for CIM most ontologies have descriptions that are small in size and easy to read. One thing to mention is that some of the ontologies do not provide descriptions for some terms. The median number of terms used is 1262 terms with half of the ontologies (Brick, CIM, DABGEO, OEO) having more than one thousand terms in use. The full list of terms and classes chosen for comparison is shown in Table 14.

Except for DABGEO all ontologies use between five and twelve ontologies. The median number of ontologies used is eight. The full list of used ontologies can be found in Table 13.

#### 4.3.4. Maintenance and Accessibility

In this category the ontologies are compared according to how well they are maintained and how accessible they are. Table 10 gives an overview over the different criteria in the category.

Table 10: A Comparison of the maintenance and accessibility of energy research ontologies. Last checked: 15.02.2023. *Open-Source (OSI)*:  $\textcircled{R}$ , *Open-Source/Active Maintenance (Accessibility/Maintenance)*:  $\checkmark$ , *Not found*:  $\times$ .

Metric \ Ontology	Maintenance and Accessibility						
	Sourcecode	License	Accessibility	Maintenance	Latest Release	Automatic Analysis	Score
Brick	2+	BSD-3- Clause	$\textcircled{R}$	$\checkmark$	V1.3 (2022-10- 12)	Problems FOOPS:0.46 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: Minor:2 Important:1	2.4

Continued on next page

Table 10: A Comparison of the maintenance and accessibility of energy research ontologies. Last checked: 15.02.2023. *Open-Source (OSI)*:  $\textcircled{R}$ , *Open-Source/Active Maintenance (Accessibility/Maintenance)*:  $\checkmark$ , *Not found*:  $\times$ . (Continued)

Metric Ontology	Maintenance and Accessibility						
	Sourcecode	License	Accessibility	Maintenance	Latest Release	Automatic Analysis	Score
CIM	2+	Apache-2.0	$\textcircled{R}$	$\checkmark$	V7.1 (2022-02-22)	Problems FOOPS:0.042 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS:unable to generate	2.4
DAB-GEO	1S	CC-BY-4.0	$\checkmark$	$\times$	V1.0 (2019-02-22)	Problems FOOPS:0.27, "F": 0.72, "A": 1.0, "I": 0.94, "R": 0.63 OOPS*: Minor:35 Important:30	1.4
EMO	1S	Not Found	$\checkmark$	$\times$	V1.0 ( $\times$ )	Problems FOOPS:0.042 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: Minor:2 Important:1	0.8
EM-KPI	2+	CC-BY-4.0	$\checkmark$	$\times$	V1.1 (2017-07-07)	Problems FOOPS:0.47 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: Minor:2 Important:1	1.6
OEO	2+	CC0-1.0	$\textcircled{R}$	$\checkmark$	V1.13.0 (2023-02-01)	Problems FOOPS:0.57 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: Minor:3 Important:1	2.4
SAR-GON	2+	CC-BY-4.0	$\checkmark$	$\times$	V1.0.1 ( $\times$ )	Problems FOOPS:0.69 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: Minor:2 Important:3	1.6
SEAS	2+	Apache-2.0	$\textcircled{R}$	$\times$	V1.1 (2017-08-29)	Problems FOOPS:0.40 "F": 0.5, "A": 2, "I": 0.0, "R": 0.0 OOPS: unable to generate	1.8

\* Excluding 11 ontologies.

Except for DABGEO and EMO the sourcecode for the ontologies is available from more than one source. For DABGEO and EMO the sourcecode is available from the respective websites. The found sources can be seen in Table 15.

Except for EMO all ontologies that are compared have a license, however half of the ontologies are not licensed through an OSI approved license. Every ontology has sourcecode openly available.

Of the eight ontologies only three, Brick, CIM, and OEO, have seen active maintenance in 2022. The other ontologies do not seem to be maintained anymore.

All of the ontologies compared have problems in the automatic analysis. Although the SARGON

ontology was tested with OOPS! when being created, not all mistakes seem to have been fixed. Furthermore, none of the other ontology creators employ automated tools to check their ontologies for errors and update them accordingly.

#### 4.3.5. Governance

In this category the ontologies are compared according to how well they are governed. Table 11 gives an overview over the different criteria in the category.

Table 11: A comparison of governance criteria between ontologies. *Not found: X*.

Metric \ Ontology	Governance			
	Funding	Governing Instances	Citations	Score
Brick	USA Saudi Arabia Denmark EU Intel Corporation	Found Brick Consortium	265	3.0
CIM	EU USA	Found IEC	3	3.0
DAB-GEO	Basque Government	X	14	0.3
EMO	EU Portugal	X	22	1.5
EM-KPI	China EU	X	34	1.5
OEO	Germany	Found OEO Steering Comitee	30	2.0
SAR-GON	Germany	X	17	0.3
SEAS	Austria Belgium Canada Finland France Germany Hungary Israel Netherlands Norway Portugal Spain Sweden Türkiye ENGIE R&D	Found <a href="https://www.maxime-lefrancois.info/">https://www.maxime-lefrancois.info/</a>	78	3.0

All of the ontologies in this comparison received funding for their creation. Most of the ontologies receive funding from governments. Brick and SEAS also received funding from the firms Intel and ENGIE R&D respectively. Only three ontologies receive funding from a single instance, the other receive funding from multiple instances. Only for half of the ontologies a governing instance was found. While for OEO and Brick the governing instance consists of a group of people, SEAS is governed by a single person and CIM is governed by IEC, a standards organization.

#### 4.4. Score Evaluation

This comparison aims to answer the question, which ontology is most suited for use in energy research postulated in subsection 1.1. This question depends on the use case of the ontology, and as such it can not be answered for every individual case. Furthermore, this comparison only partly takes the content and scope of the ontology into account and does not track the thematic coverage of an ontology.

To answer which ontology is most suited in general, the score for each category of criteria was computed for every ontology. The score was developed as objectively as possible and helps identifying the ontology best suited in general. The result of scoring each ontology can be seen in Figure 11 and Table 12.

Table 12: Accumulated scores for each ontology.

Accumulated Score	Best Practice	Practical Implementation	Maintenance and Accessibility	Governance	Ontology
10.2	2.8	3.0	2.4	2.0	OEO
9.6	2.2	2.0	2.4	3.0	Brick
9.6	1.8	3.0	1.8	3.0	SEAS
9.4	2.0	2.0	2.4	3.0	CIM
7.7	1.6	3.0	1.6	1.5	EM-KPI
6.4	2.2	2.3	1.6	0.3	SAR-GON
6.3	2.2	2.3	1.4	0.3	DAB-GEO
6.0	1.0	2.7	0.8	1.5	EMO

As can be seen from the spiderweb diagram and table, the ontologies with the highest scores are Brick, CIM, OEO and SEAS. The ontology that has the highest score overall is the Open Energy Ontology (OEO) it also has the highest score in every individual category except governance.

As all ontologies did not score full points the next section goes over some flaws in the higher scoring ontologies.

OEO is not a flawless ontology. The ontology does not make use of automated analysis tools for its validation. Furthermore, it is not as big as other ontologies like CIM and as such some concepts that are available in other ontologies might be missing in OEO. Problems with the SEAS ontology are that it is not actively maintained anymore and does not use automated analysis tools for its validation. Brick has only one ontology language format available for download and it is primarily an ontology developed for smart buildings. As such the focus of ontology engineers will not necessarily be energy relevant topics and updates in this regard might be slower than in other ontologies. The descriptions in CIM are not easily understandable for everybody. Furthermore, it is maintained through the IEC which makes it not as interactive as to what updates will be applied to it. The newest sourcecode version is also not freely available from the IEC itself and needs to be downloaded from the CIM User Groups website<sup>57</sup> and converted into the OWL format.

The ontology that is recommended by this thesis is the Open Energy Ontology (OEO). While it has some flaws it seems to be the best general choice. Furthermore, it is still actively maintained and developed. If OEO is not a good fit, Brick and CIM are also recommended for use. The SEAS ontology is not recommended as it is not actively maintained. The most important factor for choosing an ontology should be the use case as such the recommendations need to be carefully reviewed if they fit for the specific situation a researcher wants to use them in.

<sup>57</sup>See: <https://cimug.ucaiug.org/> Last checked 04.05.2023.

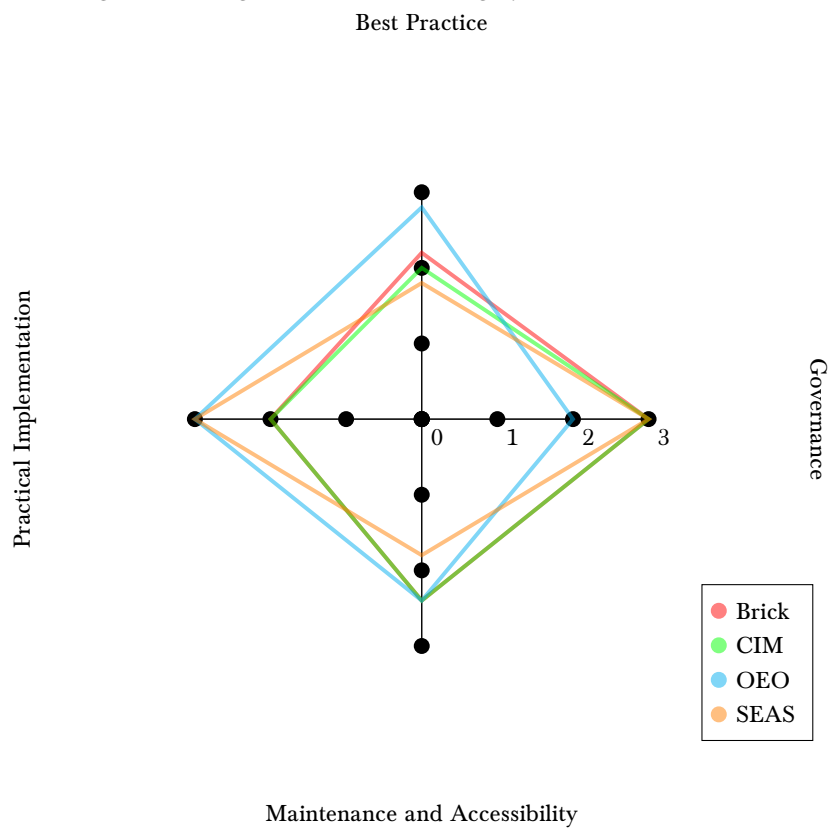
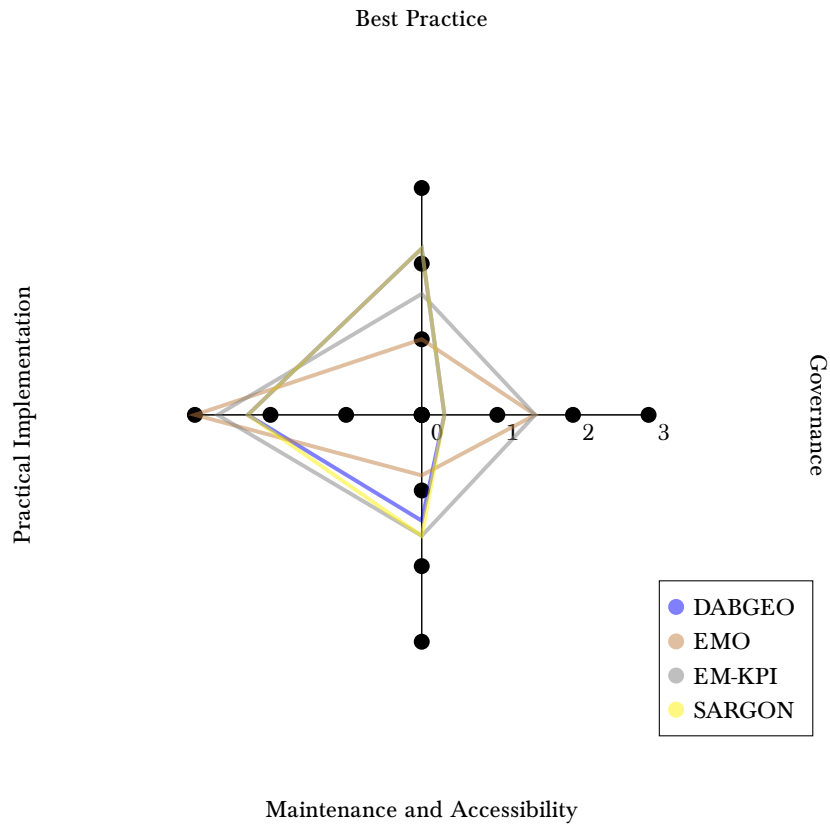


Figure 11: Spiderweb diagrams outlining the score in each category for each of the ontologies.



## 5. Development of an Ontology Picking Tool

This section describes the process of designing a tool to support researchers in choosing an ontology to use. The section is divided into looking at existing tools, planning the development of a tool, discussing the development and integrating the tool into an existing database for ontologies.

### 5.1. Existing Tools

There are some tools that can be used for choosing ontologies. The tools can be generally split into two categories, analysis tools and ontology databases.

The first category are analysis tools. Some of these tools are already described in 2.3.2. These tools, such as OOPS!, FOOPS! and FAIR-Checker, can be used to evaluate single ontologies in different categories depending on the tool and make an informed choice when deciding on an ontology.

The second category are searchable databases. These databases are tools that help searching in many ontologies at once. Some of these databases were used in the ontology selection outlined in 3.1. The databases all differ slightly in use and layout. The most basic features that is present in every different database is the ability to search for terms existing in different ontologies as well as searching for ontologies themselves. Most have some metadata about the ontologies present such as license or version. The databases can generally be split into two categories.

The first kind of database is using a walled garden approach, whereas the ontologies in the database are controlled by the institution hosting the service. A system often used in this category is the Ontology Lookup Service (OLS). The newest version is the fourth iteration and the sourcecode can be found on GitHub<sup>58</sup>.

The second approach is an open approach in which users can upload ontologies themselves. An example for this approach is LOV<sup>59</sup>.

Both approaches have different advantages and disadvantages. The walled garden approach allows a better selection and a more tight control over the ontology metadata. Furthermore, depending on the organization hosting the database, the ontologies could be seen as being more trustworthy than ontologies not in the database. This approach however, is tedious for a large amount of ontologies and inflexible. The open approach allows for an easier introduction of new ontologies, however with this approach the control over the metadata a user provides is not given and there is a possibility of wrong data being given by the ontology provider and ontologies being uploaded multiple times.

### 5.2. Planned Tool

This section first describes the problems with the existing tools and then outlines the planning of the new tool.

#### 5.2.1. Problems with Existing Tools

As explained in the text above there are already two types of tools that support researchers in choosing an ontology. In this section the positives and negatives of the tools are discussed.

The first types of tools, the analysis tools, are useful in better understanding certain aspects of ontologies as well as getting easy to comprehend data. These tools however, need to be used on every ontology individually and do not cover all aspects, that might help make an informed decision. Furthermore, they do not enable to look at the content of the ontologies in more detail. As such it is needed to look up the content with a different method and also perform each analysis for every ontology that might fit the use case.

The ontology databases are useful for searching in multiple ontologies at once. However, these databases often do not have a lot of metadata about ontologies and the search is very crude. One example is found in the TIB Terminology Service<sup>60</sup>. The homepage shown in Figure 12 is a simple search page with no further possibilities to refine the search and the search page seen in Figure 13 has four refinement options (subject, collection, term type and ontology).

There is no option to filter other important ontology metadata, e.g. filter ontologies by license. Furthermore, the currency of the ontologies and information of the ontologies is either dependent on the organization managing the database or the user base uploading the information.

<sup>58</sup>See: <https://github.com/EBISPOT/ols4> Last checked 04.05.2023.

<sup>59</sup>See: <https://lov.linkeddata.es/dataset/lov/> Last checked 04.05.2023.

<sup>60</sup>See: <https://service.tib.eu/ts4tib/index> Last checked 04.05.2023.

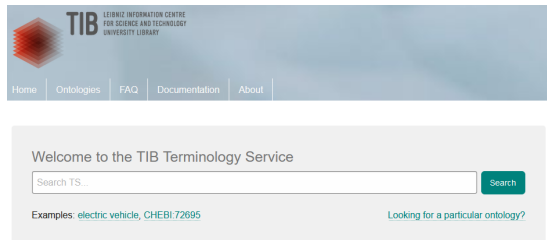


Figure 12: TIB Terminology Service Homepage. Screenshot taken 03.05.2023

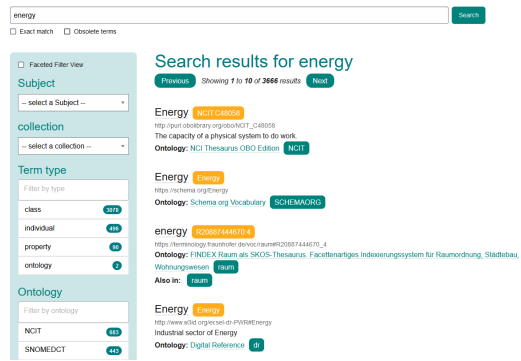


Figure 13: TIB Terminology Service search page. Screenshot taken 03.05.2023

As can be seen there are numerous tools and approaches available for choosing ontologies. These tools, however, still have some flaws as outlined in this section.

### 5.2.2. Planning

As there are some existing tools to support researchers, the plan is not to develop a standalone tool, but a tool that complements existing tools. As the goal of the tool is to support researchers in picking an ontology, the planned approach is to develop a tool that can improve the existing database tools, as they enable researchers to scan multiple ontologies at once. The biggest problem for the existing databases is the sparse availability of ontology metadata and the resulting crude search options. Therefore the planned tool tackles this problem. The planned approach is a backend tool that can be integrated into existing solutions. The backend tool uses a Representational State Transfer-Application-Programming-Interface (REST-API) that connects to a database with the data collected in this thesis. The Application-Programming-Interface (API) enables interaction with the database as well as easier integration into existing projects. The planned approach for the languages used is Python<sup>61</sup> for the API as well as PostgreSQL<sup>62</sup> for the database as these are widely used standards. The plan for the tool is summed up in the following list:

1. Backend-tool
2. Database with collected data
3. REST-API that allows database access
4. API written in Python, database in PostgreSQL

Furthermore, the tool is tested by integrating it into an existing database tool.

### 5.3. Development

The developed tool consists of two parts: a PostgreSQL database and a Python script that runs the REST-API and is also able to initialize the database.

The PostgreSQL database is relatively simple. It consists of one table containing all the criteria.

The Python script uses the module SQLAlchemy<sup>63</sup> to interact with the PostgreSQL database, the use of SQLAlchemy enables the manipulation of the database inside of the Python script without issuing SQL commands. This way the script is safe from possible user input hijacking the database if it is not cleaned properly.

Furthermore, the script uses Flask<sup>64</sup> in conjunction with webargs<sup>65</sup> to build the REST-API part of the tool. The API has three primary access points *ont*, *possible* and *names* with different capabilities.

The *ont* access point enables the user to manipulate the database and access data from the database based on a list of criteria values. The *possible* access points returns all possible values of every criteria given a subset of ontologies. The *names* access points returns a list of ontology names given a list of criteria values. The API enables the user to relay the relevant parameters via the URL. The

<sup>61</sup>See: <https://www.python.org/> Last checked 04.05.2023.

<sup>62</sup>See: <https://www.postgresql.org/> Last checked 04.05.2023.

<sup>63</sup>See: <https://www.sqlalchemy.org/> Last checked 04.05.2023.

<sup>64</sup>See: <https://flask.palletsprojects.com/en/2.3.x/> Last checked 04.05.2023.

<sup>65</sup>See: <https://github.com/marshmallow-code/webargs> Last checked 04.05.2023.

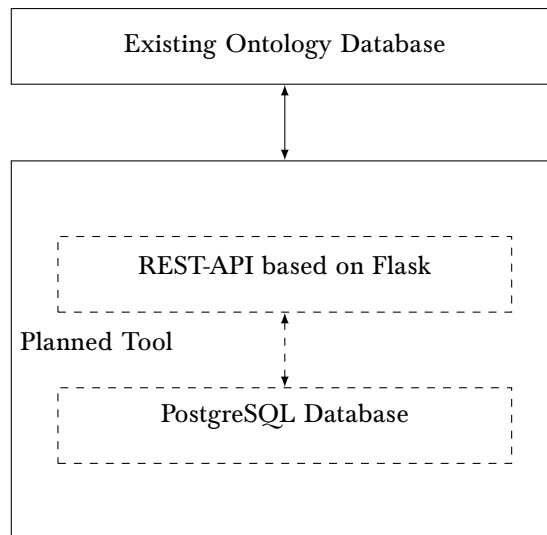


Figure 14: Sketch showing the functionality of the planned tool.

only exception is adding a new ontology, where the data is relayed via the body in JavaScript Object Notation (JSON) format. The data that is returned to the user is in JSON format. The tool is released for Open-Source access through GitHub<sup>66</sup>. Within the release is a README explaining the setup process and usage. Furthermore, the API is described with the help of SwaggerUI<sup>67</sup>. The software is available under the MIT license.

### 5.3.1. Possible Further Development

This section covers the possible further development of the tool. One further development is refining of the data access. So far it is possible to return a list of ontologies that fit a list of given criteria. One refinement would be to add a filter that returns all ontologies under or over a given criteria value, e.g. all ontologies with a governance score over 2.0. Another possible improvement would be the addition of a way to add multiple ontologies to the database at once, possibly from csv data.

## 5.4. Exemplary Integration in an Existing Database

To showcase the viability of the tool it is integrated into an existing database. As there are many options for integrating it into an existing database the choice of database is explained here.

The first choice for the integration is the TIB Terminology Service as it is associated with the University of Hanover. As the sourcecode for the Terminology Service is not Open-Source, access was requested, but it is not granted at this point. As the service is based on the OLS this service is the second choice for integration. It is Open-Source and the sourcecode is accessible. The newest version of the OLS is the fourth version, which is not yet widely used. It will be the target for the integration as it is in development.

### 5.4.1. Demonstrative Integration

The OLS version four is divided into three parts that can be developed separately:  
 dataloader: The dataloader that converts ontologies into database data for the backend  
 backend: The backend that holds the ontology data in a database  
 frontend: The frontend that serves the website a user can interact with

For this demonstrative integration, the frontend is the only part that needs to be altered. The frontend is written in React<sup>68</sup> with the styling done through Tailwind<sup>69</sup>. For development the other

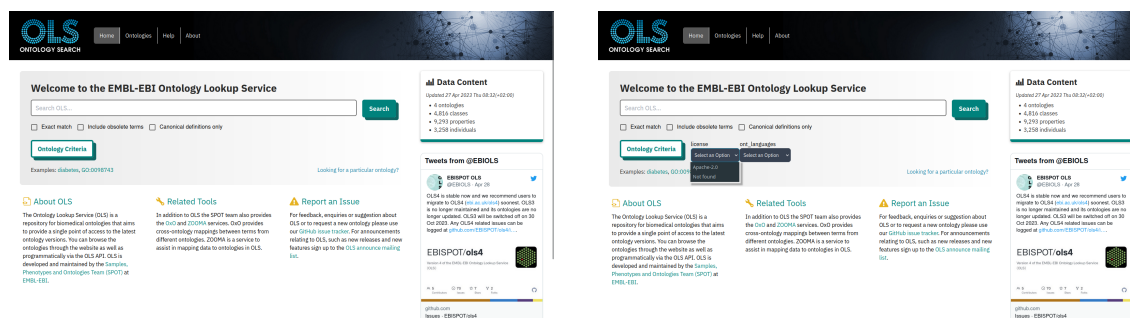
<sup>66</sup>See: <https://github.com/SupaaSchnitzel/OntMetaDatabase> The final version for this thesis is found under the tag *v1.0*. Last checked 20.05.2023.

<sup>67</sup>See: <https://swagger.io/tools/swagger-ui/> Last checked 04.05.2023.

<sup>68</sup>See: <https://react.dev/> Last checked 04.05.2023.

<sup>69</sup>See: <https://tailwindcss.com/> Last checked 04.05.2023.

parts are started up through Docker<sup>70</sup>. As can be seen in Figure 15a for this demonstration the



(a) OLS4 Homepage with added refined search. Search bar closed.

(b) OLS4 homepage with added refined search. Search bar open.

Figure 15: OLS4 homepage after integration.

homepage has an added button, that enables to user to further refine their search by using ontology criteria. By pressing the button the user is given drop down menus for two criteria as shown in Figure 15b. If criteria are chosen, the ontologies that fit the selection are shown first with the results that do not fulfill the criteria following them. For this integration, two criteria are chosen to be integrated, as integrating all criteria would lead to clutter and it also highlights that not all aspects of the tool need to be integrated.

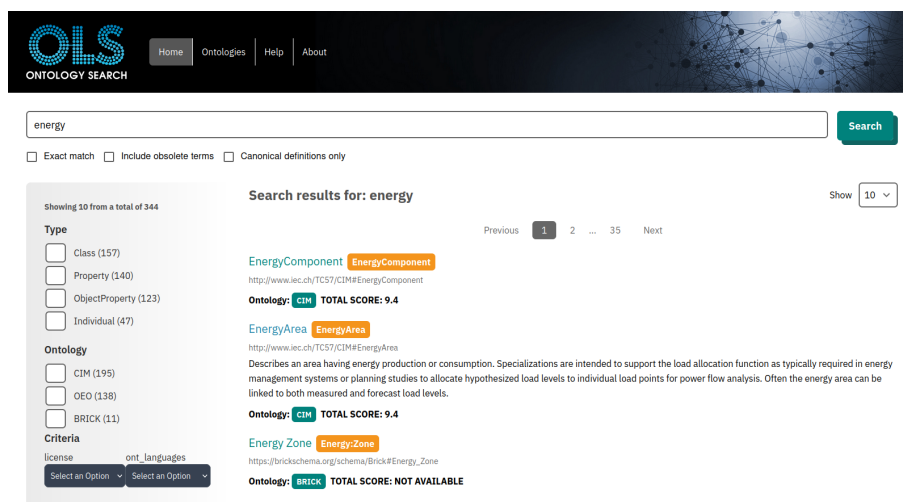


Figure 16: Modified OLS4 search results page.

Furthermore, as can be seen in Figure 16 on the search results page, the results are arranged by their accumulated score from Table 12 by default. If available, the score is also displayed under every term. The criteria selection can also be accessed on the sidebar of the search and will override the default order. The exemplary integration can be found on GitHub<sup>71</sup>.

### 5.4.2. Integration Challenges

While integrating the tool, some problems with the OLS development arose. The OLS code is not commented and therefore working with it was not as easy as it could have been. Furthermore, the frontend throws the error *ERR\_OSSL\_UNSUPPORTED* when starting it, hence, before starting the frontend, a global variable needs to be set for it to start properly: *NODE\_OPTIONS=--openssl-legacy-provider*. The cause for this seems to be that some part of the site uses OpenSSL algorithms that have fallen out of use. Another challenge is the addition of ontologies. While the process is explained

<sup>70</sup>See: <https://www.docker.com/> Last checked 04.05.2023.

<sup>71</sup>See: <https://github.com/SupaaSchnitzel/ols4-Thesis> The final version for this thesis is found under the tag *v1.0*. Last checked 20.05.2023.

in the readme file of the dataload part, it is not explained fully. Furthermore, some of the ontologies are not added as local sources because of problems that occur while loading them in the dataload process.

Another problem that occurred while integrating the tool is that the results displayed on the search results page are made available to the frontend page-by-page, meaning the results are ordered in the backend and then relayed to the frontend in parts, depending on how many results are displayed per page. For a deployable integration the backend code needs to be altered. As this is a demonstrative integration, only the frontend code was altered, thus the new results order is only applied to the results visible on the search page/available to the frontend.

#### **5.4.3. Possible Further Development**

This section covers some ideas for possible further development. One improvement that was already talked about in the problems section above is the integration of the new sorting into the backend portion of the code. A further improvement is to make the single scores in the different categories visible when hovering over the accumulated score. Furthermore, a help page or button providing information about the score is a good improvement. Possibly more criteria for refining the search could be added and the ordering of the search results could be made more user controllable.



## 6. Conclusion

This section summarizes the work done in this thesis and identifies areas for future research and development.

### 6.1. Summary

Energy is an important research field, especially in the recent years with the ongoing threat of global warming. Ontologies are a useful tool to aid in research data management. However, there are many ontologies to choose from, through this the research question: **Which ontology is suitable for a project in energy research and how can researchers be supported in choosing one?** is formed. This question is divided into four parts which are outlined in subsection 1.1.

In this thesis a comparison of energy research ontologies was realized to identify the most suitable ontology and aid researchers in selecting an ontology. For this a systematic search for ontologies in energy research was conducted in section 3. This search filtered 98 possible ontology papers down to the eight most suited to be compared answering the question which ontologies are used in energy research.

To compare the ontologies, four criteria categories were developed in section 4. These categories, best practice, practical implementation, maintenance and accessibility as well as governance contain 21 criteria in total. This answers the postulated questions, how ontologies can be compared and what the relevant criteria for a comparison are. To assist in the comparison, each of the categories were given a score through which the ontologies can be ranked. A tool was written to assist in the data gathering phase of the comparison. The Open Energy Ontology (OEO) was identified as best to be recommended on a general basis<sup>72</sup>. Thus answering the question which ontology is best suitable for a project in energy research.

Furthermore, in section 5 a tool to assist researchers in choosing an ontology was developed. The tool focuses on being integrable into existing tools as to not reinvent the wheel. This answers the question on how to support a researcher in selecting an ontology.

To showcase the use of the tool it was exemplary integrated into the existing OLS. All the tools written and data gathered in this work are openly available on GitHub and other platforms.

### 6.2. Future Work

This work showcased how to compare ontologies based on their metadata. The most obvious path for extending this work is to enlarge the set of ontologies that are compared. Adding further comparison criteria seems very promising, this is especially true for content specific criteria which were not a focus in this work. An interview of ontology engineers and researchers questioning them about important criteria to increase coverage is also worthwhile in this regard. One further area of refinement for this work is the development of the scoring into an adaptable tool that allows for a user defined score through a questionnaire.

Furthermore, the comparison process would benefit from a system that enables as many of the criteria to be gathered and analyzed automatically as possible. Ideally, such a tool would be integrated into ontology databases. Some criteria where such an automation might be possible are: license, accessibility, citations, latest release, automatic analysis, available ontology languages, number of terms used, used ontologies and used upper ontologies. The inclusion of ontology matching in the databases might also aid researchers in finding similar ontologies.

Integrating the developed tool into an ontology database to be ready for production use and further developing the tool is also a great area to focus further work on.

Further research that benefits ontology research and the ontology engineering space is better integration of the different automatic analysis tools like OOPS!, FOOPS! and FAIR-Checker into one framework that also supports the analysis of multiple ontologies at once and delivers human readable reports. While the OnToology tool mentioned in paragraph 2.3.2 is kind of similar in this regard, it is still not widely known and is missing some features that would be desirable, such as FAIR-Checker support.

A review on the landscape of energy ontology as a whole is worthwhile research, especially concerning the maintenance of ontologies. Questions such as how many ontologies are no longer maintained and how useful are not maintained ontologies come to mind.

---

<sup>72</sup>A specific recommendation depends on the use case of the ontology.





## A. Example Code for a Veterinarian Clinic

```

1 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:rdfs="http://www.
  w3.org/2000/01/rdf-schema#">
2 <rdfs:Class rdf:ID="veterinarian">
3   <rdfs:comment>
4     The class of veterinarian All veterinarian are staff members.
5   </rdfs:comment>
6   <rdfs:subClassOf rdf:resource="#staffMember"/>
7 </rdfs:Class>
8
9 <rdfs:Class rdf:ID="staffMember">
10   <rdfs:comment>The class of staff members</rdfs:comment>
11 </rdfs:Class>
12
13 <rdfs:Class rdf:ID="clinic">
14   <rdfs:comment>The class of clinics</rdfs:comment>
15 </rdfs:Class>
16
17 <rdf:Property rdf:ID="name">
18   <rdfs:comment>
19     It is a property of staff members and takes literals as values.
20   </rdfs:comment>
21   <rdfs:domain rdf:resource="#staffMember"/>
22   <rdfs:range rdf:resource="&rdf;Literal"/>
23 </rdf:Property>
24
25 <rdf:Property rdf:ID="worksat">
26   <rdfs:comment>
27     It relates only clinics to veterinarian.
28   </rdfs:comment>
29   <rdfs:domain rdf:resource="#clinic"/>
30   <rdfs:range rdf:resource="#veterinarian"/>
31 </rdf:Property>
32
33 <rdf:Property rdf:ID="email">
34   <rdfs:comment>
35     It is a property of staff members and takes literals as values.
36   </rdfs:comment> <rdfs:domain rdf:resource="#staffMember"/>
37   <rdfs:range rdf:resource="&rdf;Literal"/>
38 </rdf:Property>
39
40 <rdf:Property rdf:ID="telephone">
41   <rdfs:comment>
42     It is a property of staff members and takes literals as values.
43   </rdfs:comment> <rdfs:domain rdf:resource="#staffMember"/>
44   <rdfs:range rdf:resource="&rdf;Literal"/>
45 </rdf:Property>
46
47 <rdf:Property rdf:ID="clinicName">
48   <rdfs:comment>
49     It is a property of clinics and takes literals as values.
50   </rdfs:comment> <rdfs:domain rdf:resource="#clinic"/>
51   <rdfs:range rdf:resource="&rdf;Literal"/>
52 </rdf:Property>
53 </rdf:RDF>

```

Figure 17: Full RDFS code for the example given in subsection 2.3.1.

```

1 <?xml version="1.0"?>
2 <rdf:RDF xmlns="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#"
3   xml:base="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic"
4   xmlns:owl="http://www.w3.org/2002/07/owl#"
5   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6   xmlns:xml="http://www.w3.org/XML/1998/namespace"
7   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
8   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
9   xmlns:petclinic="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#">
10 <owl:Ontology rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
11   petclinic">
12   <owl:versionIRI rdf:resource="http://www.semanticweb.org/delph/ontologies
13     /2023/0/petclinic/1.0.0"/>
14 </owl:Ontology>
15 <!--
16 // Object Properties
17 //
18 ////////////////////////////////////////////////////
19 -->
20 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#works_at -->
21 <owl:ObjectProperty rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
22   petclinic#works_at">
23   <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#
24     topObjectProperty"/>
25   <rdfs:domain rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
26     petclinic#clinic"/>
27   <rdfs:range rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
28     petclinic#staffMember"/>
29 </owl:ObjectProperty>
30 <!--
31 // Data properties
32 //
33 ////////////////////////////////////////////////////
34 -->
35 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#clinicName -->
36 <owl:DatatypeProperty rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
37   petclinic#clinicName">
38   <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"
39     />
40   <rdfs:domain rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
41     petclinic#clinic"/>
42   <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
43   <owl:propertyDisjointWith rdf:resource="http://www.semanticweb.org/delph/
44     ontologies/2023/0/petclinic#name"/>
45 </owl:DatatypeProperty>
46 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#email -->

```

Figure 18: First part of the RDFS/OWL code for the example given in subsection 2.3.1.

```

1 <owl:DatatypeProperty rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
2 petclinic#email">
3   <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"
4   />
5   <rdfs:domain rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
6 petclinic#staffMember"/>
7   <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
8 </owl:DatatypeProperty>
9 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#name -->
10 <owl:DatatypeProperty rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
11 petclinic#name">
12   <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"
13   />
14   <rdfs:domain rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
15 petclinic#staffMember"/>
16   <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
17 </owl:DatatypeProperty>
18 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#telephone -->
19 <owl:DatatypeProperty rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
20 petclinic#telephone">
21   <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"
22   />
23   <rdfs:domain rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
24 petclinic#staffMember"/>
25   <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
26 </owl:DatatypeProperty>
27 <!--
28 // Classes
29 //
30 //
31 -->
32 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#clinic -->
33 <owl:Class rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#
34 clinic">
35   <rdfs:comment>The class of clinics</rdfs:comment>
36 </owl:Class>
37 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#staffMember -->
38 <owl:Class rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#
39 staffMember">
40   <rdfs:comment>The class of staff members</rdfs:comment>
41 </owl:Class>
42 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#veterinarian -->
43 <owl:Class rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#
44 veterinarian">
45   <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/delph/ontologies
46 /2023/0/petclinic#staffMember"/>
47   <rdfs:comment>The class of veterinarian All veterinarian are staff members.</
48 rdfs:comment>
49 </owl:Class>
50 <!--

```

Figure 19: Second part of the RDFS/OWL code for the example given in subsection 2.3.1.

```

1 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
2 //
3 // Individuals
4 //
5 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
6 -->
7 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#Help_Pets_Inc -->
8 <owl:NamedIndividual rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
9 petclinic#Help_Pets_Inc">
10   <rdf:type rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
11 petclinic#clinic"/>
12   <clinicName>Help Pets Inc</clinicName>
13 </owl:NamedIndividual>
14 <!-- http://www.semanticweb.org/delph/ontologies/2023/0/petclinic#Jane_Doe -->
15 <owl:NamedIndividual rdf:about="http://www.semanticweb.org/delph/ontologies/2023/0/
16 petclinic#Jane_Doe">
17   <rdf:type rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
18 petclinic#veterinarian"/>
19   <works_at rdf:resource="http://www.semanticweb.org/delph/ontologies/2023/0/
20 petclinic#Help_Pets_Inc"/>
21   <email rdf:datatype="http://www.w3.org/2000/01/rdf-schema#Literal">Jane.
22   Doe@helpetsinc.com</email>
23   <name>Jane Doe</name>
24   <telephone rdf:datatype="http://www.w3.org/2000/01/rdf-schema#Literal">1505 -644
25   -1287</telephone>
26 </owl:NamedIndividual>
27 <!--
28 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
29 //
30 // General axioms
31 //
32 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
33 -->
34 <rdf:Description>
35   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDifferent"/>
36   <owl:distinctMembers rdf:parseType="Collection">
37     <rdf:Description rdf:about="http://www.semanticweb.org/delph/ontologies
38 /2023/0/petclinic#Help_Pets_Inc"/>
39     <rdf:Description rdf:about="http://www.semanticweb.org/delph/ontologies
40 /2023/0/petclinic#Jane_Doe"/>
41   </owl:distinctMembers>
42 </rdf:Description>
43 </rdf:RDF>

```

Figure 20: Third part of the RDFS/OWL code for the example given in subsection 2.3.1.

## B. Tables with Data from the Analysis

Table 13: Ontologies used by each ontology of the comparison.

Ontology	Used Ontologies
Brick	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> dcterms: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> bacnet: <a href="http://data.ashrae.org/bacnet/2020">http://data.ashrae.org/bacnet/2020</a> sh: <a href="http://www.w3.org/ns/shacl">http://www.w3.org/ns/shacl</a> sdo: <a href="http://schema.org/">http://schema.org/</a> qudt: <a href="http://qudt.org/schema/qudt/">http://qudt.org/schema/qudt/</a> skos: <a href="http://www.w3.org/2004/02/skos/core">http://www.w3.org/2004/02/skos/core</a>
CIM	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>
DABGEO	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> No name0: WaterHeatingWhiteGoods/ terms: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> ns: <a href="http://creativecommons.org/ns">http://creativecommons.org/ns</a> Battery: <a href="https://www.purl.org/gepss/resource/Battery/">https://www.purl.org/gepss/resource/Battery/</a> dc: <a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a> saref4ee: <a href="https://w3id.org/saref4ee">https://w3id.org/saref4ee</a> enaeq: <a href="http://www.purl.org/oema/enaeq/">http://www.purl.org/oema/enaeq/</a> saref: <a href="https://w3id.org/saref">https://w3id.org/saref</a> dogont: <a href="http://elite.polito.it/ontologies/dogont.owl">http://elite.polito.it/ontologies/dogont.owl</a> swrlb: <a href="http://www.w3.org/2003/11/swrlb">http://www.w3.org/2003/11/swrlb</a> EnergyResourceOntology: <a href="https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/">https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/</a> EnergyResourceOntology.owl units: <a href="http://www.purl.org/oema/units/">http://www.purl.org/oema/units/</a> BuildingOntology: <a href="https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/BuildingOntology.owl">https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/BuildingOntology.owl</a> swrlx: <a href="http://www.w3.org/2003/11/swrlx">http://www.w3.org/2003/11/swrlx</a> owl2: <a href="http://www.w3.org/2006/12/owl2">http://www.w3.org/2006/12/owl2</a> prosg: <a href="http://www.purl.org/net/v3/prosg">http://www.purl.org/net/v3/prosg</a> efficiency: <a href="http://www.ida.liu.se/projects/semtech/schemas/energy/2013/09/efficiency.owl">http://www.ida.liu.se/projects/semtech/schemas/energy/2013/09/efficiency.owl</a> swrl: <a href="http://www.w3.org/2003/11/swrl">http://www.w3.org/2003/11/swrl</a> cpannotationschema: <a href="http://www.ontologydesignpatterns.org/schemas/cpannotationschema.owl">http://www.ontologydesignpatterns.org/schemas/cpannotationschema.owl</a> water: <a href="http://vocab.deri.ie/water">http://vocab.deri.ie/water</a>

Continued on next page

Table 13: Ontologies used by each ontology of the comparison. (Continued)

Ontology	Used Ontologies
DABGEO	<p>mirabel: <a href="http://sites.google.com/site/smartappliancesproject/ontologies/mirabel">http://sites.google.com/site/smartappliancesproject/ontologies/mirabel</a>  BOnSAI: <a href="http://lpis.csd.auth.gr/ontologies/bonsai/BOnSAI.owl">http://lpis.csd.auth.gr/ontologies/bonsai/BOnSAI.owl</a>  om-1: <a href="http://www.wurvoc.org/vocabularies/om-1.8/">http://www.wurvoc.org/vocabularies/om-1.8/</a>  time: <a href="http://www.w3.org/2006/time">http://www.w3.org/2006/time</a>  terms1: <a href="http://purl.org/dc/elements/1.1/terms">http://purl.org/dc/elements/1.1/terms</a>  WeatherOntology:  <a href="https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/WeatherOntology.owl">https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/WeatherOntology.owl</a>  SEMANCO: <a href="http://semanco02.hs-albsig.de/repository/ontology-releases/eu/semanco/ontology/SEMANCO/SEMANCO.owl">http://semanco02.hs-albsig.de/repository/ontology-releases/eu/semanco/ontology/SEMANCO/SEMANCO.owl</a>  schema: <a href="http://www.disit.org/km4city/schema">http://www.disit.org/km4city/schema</a>  BuildingOntologySharedVocabulary:  <a href="https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/BuildingOntologySharedVocabulary.owl">https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/BuildingOntologySharedVocabulary.owl</a>  rooms: <a href="http://vocab.deri.ie/rooms">http://vocab.deri.ie/rooms</a>  EnergyConsumption:  <a href="http://smartcity.linkeddata.es/lcc/ontology/EnergyConsumption">http://smartcity.linkeddata.es/lcc/ontology/EnergyConsumption</a>  infrastructure: <a href="http://www.purl.org/oema/infrastructure/">http://www.purl.org/oema/infrastructure/</a>  ontology: <a href="http://dbpedia.org/ontology/">http://dbpedia.org/ontology/</a>  EnergyAndResourceOntology:  <a href="https://eps.mondragon.edu/MUEOnt/EnergyAndResourceOntology/">https://eps.mondragon.edu/MUEOnt/EnergyAndResourceOntology/</a>  externalfactors: <a href="http://www.purl.org/oema/externalfactors/">http://www.purl.org/oema/externalfactors/</a>  Building: <a href="http://dbpedia.org/ontology/Building/">http://dbpedia.org/ontology/Building/</a>  skos: <a href="http://www.w3.org/2004/02/skos/core">http://www.w3.org/2004/02/skos/core</a>  ontology2: <a href="http://dbpedia.org/ontology/">http://dbpedia.org/ontology/</a>  SUMO: <a href="http://www.ontologyportal.org/SUMO.owl">http://www.ontologyportal.org/SUMO.owl</a></p>
DABGEO	<p>owlapi: <a href="http://www.semanticweb.org/owlapi">http://www.semanticweb.org/owlapi</a>  timezone: <a href="http://www.w3.org/2006/timezone">http://www.w3.org/2006/timezone</a>  enaeq5: <a href="http://www.purl.org/oema/enaeq/100-1000">http://www.purl.org/oema/enaeq/100-1000</a>  enaeq4: <a href="http://www.purl.org/oema/enaeq/0-10">http://www.purl.org/oema/enaeq/0-10</a>  enaeq3: <a href="http://www.purl.org/oema/enaeq/10-100">http://www.purl.org/oema/enaeq/10-100</a>  enaeq2: <a href="http://www.purl.org/oema/enaeq/1000-10000">http://www.purl.org/oema/enaeq/1000-10000</a>  CoDAMoS: <a href="http://lpis.csd.auth.gr/ontologies/CoDAMoS/CoDAMoS.owl">http://lpis.csd.auth.gr/ontologies/CoDAMoS/CoDAMoS.owl</a>  ActorOntology:  <a href="https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/ActorOntology.owl">https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/ActorOntology.owl</a>  sgstakeholders: <a href="http://www.purl.org/oema/sgstakeholders/">http://www.purl.org/oema/sgstakeholders/</a>  muo: <a href="http://purl.oclc.org/NET/muo/muo">http://purl.oclc.org/NET/muo/muo</a>  ns1: <a href="http://creativecommons.org/ns">http://creativecommons.org/ns</a>  eu: <a href="http://socsem.open.ac.uk/ontologies/eu">http://socsem.open.ac.uk/ontologies/eu</a>  core: <a href="http://purl.org/muto/core">http://purl.org/muto/core</a>  adms: <a href="http://www.w3.org/ns/adms">http://www.w3.org/ns/adms</a>  bp: <a href="http://vocab.deri.ie/bp">http://vocab.deri.ie/bp</a>  dcterms: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a>  vann: <a href="http://purl.org/vocab/vann/">http://purl.org/vocab/vann/</a>  foaf: <a href="http://xmlns.com/foaf/0.1/">http://xmlns.com/foaf/0.1/</a>  pen: <a href="http://vocab.deri.ie/pen">http://vocab.deri.ie/pen</a>  error: <a href="http://org.semanticweb.owlapi/error">http://org.semanticweb.owlapi/error</a>  unit: <a href="http://www.w3.org/2007/ont/unit">http://www.w3.org/2007/ont/unit</a>  prov: <a href="http://www.w3.org/ns/prov">http://www.w3.org/ns/prov</a>  schema2: <a href="http://www.disit.org/km4city/schema/">http://www.disit.org/km4city/schema/</a>  pao: <a href="http://www.purl.org/oema/pao/">http://www.purl.org/oema/pao/</a></p>

Continued on next page

Table 13: Ontologies used by each ontology of the comparison. (Continued)

Ontology	Used Ontologies
EMO	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>
EM-KPI	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> ns1: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> ns2: <a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>
OEO	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> dc: <a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a> uo: <a href="http://purl.obolibrary.org/obo/uo">http://purl.obolibrary.org/obo/uo</a> obo: <a href="http://purl.obolibrary.org/obo/">http://purl.obolibrary.org/obo/</a> foaf: <a href="http://xmlns.com/foaf/0.1/">http://xmlns.com/foaf/0.1/</a> terms: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> protege: <a href="http://protege.stanford.edu/plugins/owl/protege">http://protege.stanford.edu/plugins/owl/protege</a> oboInOwl: <a href="http://www.geneontology.org/formats/oboInOwl">http://www.geneontology.org/formats/oboInOwl</a>
SARGON	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> ns1: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a>
SEAS	owl: <a href="http://www.w3.org/2002/07/owl">http://www.w3.org/2002/07/owl</a> rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns">http://www.w3.org/1999/02/22-rdf-syntax-ns</a> rdfs: <a href="http://www.w3.org/2000/01/rdf-schema">http://www.w3.org/2000/01/rdf-schema</a> xsd: <a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a> xml: <a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a> dcterms: <a href="http://purl.org/dc/terms/">http://purl.org/dc/terms/</a> voaf: <a href="http://purl.org/vocommons/voaf">http://purl.org/vocommons/voaf</a> vann: <a href="http://purl.org/vocab/vann/">http://purl.org/vocab/vann/</a> foaf: <a href="http://xmlns.com/foaf/0.1/">http://xmlns.com/foaf/0.1/</a>

Table 14: Classes and properties of the compared ontologies as well as the classes chosen for Description Size and Quality. Behind the chosen classes is their assessment. S:Small, M:Medium L:Large E:Easy, N:Neutral, H:Hard. ND:No Description

<b>Ontology</b>	<b>Number of Terms</b>	<b>Chosen Classes</b>
Brick	Classes: 1454 Annotation properties: 58 Object properties:88 Data properties:23 Total properties: 169	Air Alarm <i>ND</i> Mixed Air Humidity Setpoint <i>S, E</i> Liquid CO2 <i>S, E</i> Surveillance Camera <i>ND</i> Touchpanel <i>S, E</i>
CIM	Classes:1933 Annotation properties:0 Data properties:2216 Object properties:6773 Total properties:8989	ShuntCompensatorDynamicData <i>S, H</i> ConstraintResults <i>S, H</i> ExcSCRX <i>S, H</i> MktGeneratingUnit <i>S, E</i> ChargeProfile <i>S, E</i>
DABGEO	Classes:1965 Annotation properties:84 Data properties:198 Object properties:276 Total properties:558	SpaceCoolingGenerationSubsystem <i>S, E</i> HEMS <i>S, E</i> ExtremeHeat <i>ND</i> FinancialService <i>S, E</i> JobSeekersAllowance <i>S, E</i>
EMO	Classes:15 Annotation properties:0 Data properties:15 Object properties:22 Total properties:37	Price <i>S, E</i> Player <i>S, E</i> Power <i>S, E</i> Area <i>S, E</i> Offer <i>S, E</i>
EM-KPI	Classes:134 Annotation properties:9 Data properties:13 Object properties:125 Total properties:147	EnergySupplyFromSiteProduction <i>S, E</i> Equation <i>S, E</i> Interval <i>S, E</i> EnergySupply <i>S, E</i> BuildingType <i>S, E</i>
OEO	Classes:1416 Annotation properties:100 Data properties:1 Object properties:118 Total properties:219	industrial process <i>M, E</i> parameterisation <i>S, M</i> heat pump <i>S, E</i> watercraft <i>S, E</i> day <i>S, E</i>
SARGON	Classes:150 Annotation properties:7 Data properties:56 Object properties:53 Total properties:116	Profile <i>M, E</i> Power_meter <i>ND</i> Channel <i>ND</i> OpenState <i>S, E</i> BuildingRelated <i>S, M</i>
SEAS	Classes:562 Annotation properties:18 Data properties:13 Object properties:308 Total properties:339	Balancing <i>S, E</i> PV Energy Generation Forecasting <i>S, E</i> Boolean Actuating <i>S, E</i> Area Property <i>S, E</i> Unrenewable Energy Form <i>S, E</i>



Table 15: Found Sourcecode for each ontology. The sourcecode is not necessarily official.

Ontology	Sourcecode
Brick	Website: <a href="https://brickschema.org/">https://brickschema.org/</a> Github: <a href="https://github.com/BrickSchema/Brick">https://github.com/BrickSchema/Brick</a>
CIM	Website: <a href="https://cimug.ucaiug.org/CIM%20Model%20Releases/Forms/AllItems.aspx">https://cimug.ucaiug.org/CIM%20Model%20Releases/Forms/AllItems.aspx</a> Github: <a href="https://github.com/CIMug-org/IEC-CIM-Ontology">https://github.com/CIMug-org/IEC-CIM-Ontology</a> , <a href="https://github.com/rwl/PyCIM">https://github.com/rwl/PyCIM</a> , <a href="https://github.com/sogno-platform/cimpy">https://github.com/sogno-platform/cimpy</a>
DABGEO	Website: <a href="https://innoweb.mondragon.edu/ontologies/dabgeo/">https://innoweb.mondragon.edu/ontologies/dabgeo/</a>
EMO	Website: <a href="http://www.massem.gecad.isep.ipp.pt/ontologies/electricity-markets.owl">http://www.massem.gecad.isep.ipp.pt/ontologies/electricity-markets.owl</a>
EM-KPI	Website: <a href="http://energy.linkeddata.es/em-kpi/ontology/index-en.html">http://energy.linkeddata.es/em-kpi/ontology/index-en.html</a> Github: <a href="https://github.com/nandana/em-kpi-ontology">https://github.com/nandana/em-kpi-ontology</a>
OEO	Website: <a href="https://openenergy-platform.org/ontology/oeo">https://openenergy-platform.org/ontology/oeo</a> Github: <a href="https://github.com/OpenEnergyPlatform/ontology">https://github.com/OpenEnergyPlatform/ontology</a> TIB Terminology Service: <a href="https://service.tib.eu/ts4tib/ontologies/oeo">https://service.tib.eu/ts4tib/ontologies/oeo</a>
SARGON	Website: <a href="https://sargon-n5geh.netlify.app/">https://sargon-n5geh.netlify.app/</a> Github: <a href="https://github.com/N5GEH/SARGON-Website">https://github.com/N5GEH/SARGON-Website</a>
SEAS	Website: <a href="https://ci.mines-stetienne.fr/seas/index.html">https://ci.mines-stetienne.fr/seas/index.html</a> Github: <a href="https://github.com/thSMARTenergy/seas">https://github.com/thSMARTenergy/seas</a> LOV: <a href="https://lov.linkeddata.es/dataset/lov/vocabs/seas">https://lov.linkeddata.es/dataset/lov/vocabs/seas</a>



## Glossary

*API* Application-Programming-Interface. 34, 35

*BFO* Basic Formal Ontology. 19

*BOnSAI* Smart Building Ontology for Ambient Intelligence. 16

*Brick* Brick Ontology. 18, 25–31, 45, 48, 49

*CIM* Common Information Model. 17–19, 26–31, 45, 48, 49

*DABGEO* Domain Analysis-Based Global Energy Ontology. 17–19, 25–31, 45, 46, 48, 49

*DAML+OIL* Language for describing ontologies. Jointly developed by U.S. and European research groups based on the languages DAML-ONT and Notation3 [13]. 7

*domain ontologies* Ontologies that describe concept in a specific domain of knowledge e.g., energy.. 4

*EM-KPI* Energy Management Key Performance Indicator Ontology. 17–19, 26–31, 47–49

*EMO* Electricity Markets Ontology. 18, 19, 26, 28–31, 47–49

*ETSI* European Telecommunications Standards Institute. 19, 20

*FAIR* findable, accessible, interoperable and reusable. 1, 13

*FAIR-Checker* Fair-Checker. 13, 24, 25, 33, 39

*FOOPS!* Ontology Pitfall Scanner for FAIR. 12, 13, 24–26, 33, 39

*HTML* hypertext markup language. 5

*HTML5* HTML Version 5.X.. 5

*IEC* International Electrotechnical Commission. 27, 30, 31

*IEEE* Institute of Electrical and Electronics Engineers. 16

*IoT* Internet of Things. 20

*IRI* International Resource Identifier. 6, 8, 10

*ISO* International Organization for Standardization. 3

*JSON* JavaScript Object Notation. 35

*JSON-LD* JSON based language for serialization of linked data, can be used as syntax for RDF. Is a superset of RDF. [27].. 5, 11

*LOGSPACE* “The class of problems solvable by a deterministic algorithm using space that is at most logarithmic in the size of the input (i.e., roughly  $\log(n)$ , for  $n$  the size of the input and  $c$  a constant).” [49, Chapter 5 Computational properties]. 9

*LOV* Linked Open Vocabularies. 16, 24, 33, 49

*microdata* A format, that like RDFa can be used to extend web formats like HTML with machine readability and metadata.. 5

*N-Triples* Syntax used to describe RDF graphs, line-based. Subset of Turtle [56].. 5, 11

*NISO* National Information Standards Organization. 3

- Notation3* Language developed as syntax for RDF with human-readability in mind. As a superset of RDF it has more capabilities [43].. 5, 11
- OEO* Open Energy Ontology. v, 4, 17–19, 25–31, 39, 47–49
- OEP* Open Energy Platform. 19
- OLS* Ontology Lookup Service. 24, 33, 35, 36, 39
- OOPS!* Ontology Pitfall Scanner!. 10, 12, 13, 23, 24, 26, 30, 33, 39
- ORKG* Open Research Knowledge Graph. 25
- OSI* open source initiative. 24, 29
- OWL* Web Ontology Language. 5, 7–11, 13, 31, 42–44
- OWL1* The earlier version of OWL first specified in 2004 [3, pp. 113-135].. 7–10
- OWL2* Superset of OWL1. Developed in 2009 by W3C to add more capabilities[49].. 7–12
- PRISMA* Preferred Reporting Items for Systematic Reviews and Meta-Analyses. v, 15
- RDF* Resource Description Framework. 4–11
- RDFa* Stands for RDF in attributes. It is a W3C recommendation for extending HTML/HTML5 and other formats like XHTML with machine readability and metadata. . 5
- RDFS* Resource Description Framework Schema. 5–9, 41–44
- RESTAPI* Representational State Transfer-Application-Programming-Interface. 34
- SAREF* Smart Application Reference Ontology. 19, 20
- SARGON* Smart Energy Domain Ontology. 17–20, 27–31, 47–49
- SEAS* Smart Energy Aware Systems Ontology. 18, 20, 25, 27–31, 47–49
- SGML* standard generalized markup language. 5
- SPARQL* SPARQL Protocol And RDF Query Language. 6
- TIB* Leibniz-Informationszentrum Technik und Naturwissenschaften und Universitätsbibliothek. 24, 33–35, 49
- Turtle* Developed as a more compact subset of Notation3. Superset of N-Triples [58].. 5, 11
- upper ontologies* Ontologies that describe very general concept using terms that are not exclusive to any domain of knowledge.. 4
- URI* Uniform Resource Identifier. 6, 10
- URL* Uniform Resource Locator. 6, 34
- W3C* International consortium with the goal of developing an open standard for web technologies [2].. 3, 5–7, 9, 10, 12
- XHTML* Markup language based on HTML Version 4. W3C Standard that can be seen as a mix of HTML and XML.. 5
- XML* extensible markup language. 4–11

## A. References

- [1] *A Semantic Ontology for Representing and Quantifying Energy Flexibility of Buildings* | Elsevier Enhanced Reader. DOI: 10.1016/j.adapen.2022.100113. URL: <https://reader.elsevier.com/reader/sd/pii/S2666792422000312?token=8CA5A402858D8D346450AB717689A4103C9B0FD70C471CD019E92AB1147567BF4A414022E420&originRegion=eu-west-1&originCreation=20230113180257> (visited on 01/13/2023).
- [2] *About W3C*. URL: <https://www.w3.org/Consortium/> (visited on 12/13/2022).
- [3] G. Antoniou and Frank Van Harmelen. *A Semantic Web Primer*. 2nd ed. Cooperative Information Systems. Cambridge, Mass: MIT Press, 2008. 264 pp. ISBN: 978-0-262-01242-3.
- [4] Robert Arp, Barry Smith, and Andrew D. Spear. *Building Ontologies with Basic Formal Ontology*. Cambridge, Massachusetts: Massachusetts Institute of Technology, 2015. 220 pp. ISBN: 978-0-262-52781-1.
- [5] Bharathan Balaji et al. "Brick: Towards a Unified Metadata Schema For Buildings." In: *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. BuildSys '16: The 3rd ACM International Conference on Systems for Energy-Efficient Built Environments. Palo Alto CA USA: ACM, Nov. 16, 2016, pp. 41–50. ISBN: 978-1-4503-4264-3. DOI: 10.1145/2993422.2993577. URL: <https://dl.acm.org/doi/10.1145/2993422.2993577> (visited on 11/03/2022).
- [6] A L Beck and V Rai. "Solar Soft Cost Ontology: A Review of Solar Soft Costs." In: *Progress in Energy* 2.1 (Dec. 27, 2019), p. 012001. ISSN: 2516-1083. DOI: 10.1088/2516-1083/ab59be. URL: <https://iopscience.iop.org/article/10.1088/2516-1083/ab59be> (visited on 11/03/2022).
- [7] Meisam Booshehri et al. "Introducing the Open Energy Ontology: Enhancing Data Interpretation and Interfacing in Energy Systems Analysis." In: *Energy and AI* 5 (Sept. 2021), p. 100074. ISSN: 26665468. DOI: 10.1016/j.egyai.2021.100074. URL: <https://linkinghub.elsevier.com/retrieve/pii/S2666546821000288> (visited on 11/03/2022).
- [8] Paolo Brizzi et al. "Towards an Ontology Driven Approach for Systems Interoperability and Energy Management in the Smart City." In: *2016 International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*. 2016 International Multidisciplinary Conference on Computer and Energy Science (SpliTech). Split, Croatia: IEEE, July 2016, pp. 1–7. DOI: 10.1109/SpliTech.2016.7555948. URL: <http://ieeexplore.ieee.org/document/7555948/> (visited on 11/03/2022).
- [9] Neil P. Chue Hong et al. "FAIR Principles for Research Software (FAIR4RS Principles)." In: (2021). In collab. with Alexander Struck et al. DOI: 10.15497/RDA00065. URL: <https://rd-alliance.org/group/fair-research-software-fair4rs-wg/outcomes/fair-principles-research-software-fair4rs> (visited on 05/08/2023).
- [10] Edward Corry et al. "A Performance Assessment Ontology for the Environmental and Energy Management of Buildings." In: *Automation in Construction* 57 (Sept. 2015), pp. 249–259. ISSN: 09265805. DOI: 10.1016/j.autcon.2015.05.002. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0926580515001041> (visited on 11/03/2022).
- [11] Javier Cuenca, Felix Larrinaga, and Edward Curry. "A Unified Semantic Ontology for Energy Management Applications." In: (), p. 12.
- [12] Javier Cuenca, Felix Larrinaga, and Edward Curry. "DABGEO: A Reusable and Usable Global Energy Ontology for the Energy Domain." In: *Journal of Web Semantics* (Mar. 2020), p. 100550. ISSN: 15708268. DOI: 10.1016/j.websem.2020.100550. URL: <https://linkinghub.elsevier.com/retrieve/pii/S1570826820300020> (visited on 11/03/2022).
- [13] *DAML+OIL (March 2001) Reference Description*. URL: <https://www.w3.org/TR/daml+oil-reference/> (visited on 12/14/2022).
- [14] Laura Daniele et al. "Interoperability for Smart Appliances in the IoT World." In: *The Semantic Web – ISWC 2016*. Ed. by Paul Groth et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 21–29. ISBN: 978-3-319-46547-0. DOI: 10.1007/978-3-319-46547-0\_3.
- [15] Houssemeddine Degha et al. "Onto-SB: Human Profile Ontology for Energy Efficiency in Smart Building." In: *2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)*. 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS). Oct. 2018, pp. 1–8. DOI: 10.1109/PAIS.2018.8598509.
- [16] Aravind Devanand et al. "OntoPowSys: A Power System Ontology for Cross Domain Interactions in an Eco Industrial Park." In: *Energy and AI* 1 (Aug. 1, 2020), p. 100008. ISSN: 2666-5468. DOI: 10.1016/j.egyai.2020.100008. URL: <https://www.sciencedirect.com/science/article/pii/S2666546820300082> (visited on 12/07/2022).
- [17] *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. URL: <https://www.w3.org/TR/xml/> (visited on 12/13/2022).

- [18] Stephan Ferenz et al. “An Open Digital Platform to Support Interdisciplinary Energy Research and Practice—Conceptualization.” In: *Energies* 15.17 (Sept. 2, 2022), p. 6417. ISSN: 1996-1073. DOI: 10.3390/en15176417. URL: <https://www.mdpi.com/1996-1073/15/17/6417> (visited on 05/08/2023).
- [19] Daniel Garijo, Oscar Corcho, and Maria Poveda-Villalon. “FOOPS!: An Ontology Pitfall Scanner for the FAIR Principles.” In: ().
- [20] Syed Gillani, Frederique Laforest, and Gauthier Picard. “A Generic Ontology for Prosumer-Oriented Smart Grid.” In: (), p. 6.
- [21] Tonci Grubic and Ip-Shing Fan. “Supply Chain Ontology: Review, Analysis and Synthesis.” In: *Computers in Industry. Semantic Web Computing in Industry* 61.8 (Oct. 1, 2010), pp. 776–786. ISSN: 0166-3615. DOI: 10.1016/j.compind.2010.05.006. URL: <https://www.sciencedirect.com/science/article/pii/S0166361510000552> (visited on 01/17/2023).
- [22] Maliheh Haghgoo et al. “SARGON – Smart Energy Domain Ontology.” In: *IET Smart Cities* 2.4 (Dec. 2020), pp. 191–198. ISSN: 2631-7680, 2631-7680. DOI: 10.1049/iet-smc.2020.0049. URL: <https://onlinelibrary.wiley.com/doi/10.1049/iet-smc.2020.0049> (visited on 11/03/2022).
- [23] Wilhelm Hasselbring et al. “From FAIR Research Data toward FAIR and Open Research Software.” In: *it - Information Technology* 62.1 (Feb. 25, 2020), pp. 39–47. ISSN: 2196-7032, 1611-2776. DOI: 10.1515/itit-2019-0040. URL: <https://www.degruyter.com/document/doi/10.1515/itit-2019-0040/html> (visited on 11/03/2022).
- [24] Tianzhen Hong et al. “An Ontology to Represent Energy-Related Occupant Behavior in Buildings. Part I: Introduction to the DNAs Framework.” In: *Building and Environment* 92 (Oct. 1, 2015), pp. 764–777. ISSN: 0360-1323. DOI: 10.1016/j.buildenv.2015.02.019. URL: <https://www.sciencedirect.com/science/article/pii/S0360132315000761> (visited on 01/17/2023).
- [25] Tianzhen Hong et al. “An Ontology to Represent Energy-Related Occupant Behavior in Buildings. Part II: Implementation of the DNAs Framework Using an XML Schema.” In: *Building and Environment* 94 (Dec. 1, 2015), pp. 196–205. ISSN: 0360-1323. DOI: 10.1016/j.buildenv.2015.08.006. URL: <https://www.sciencedirect.com/science/article/pii/S0360132315300871> (visited on 01/18/2023).
- [26] IEC IEC. *61970-301: Energy management system application program interface (EMS-API)-Part 301: Common Information Model (CIM) Base*. Tech. rep. Technical report, IEC-International Electrotechnical Commission, 2003.
- [27] *JSON-LD 1.1*. URL: <https://www.w3.org/TR/json-ld/> (visited on 12/13/2022).
- [28] Fadime Kaya and Jaap Gordijn. “DECENT: An Ontology for Decentralized Governance in the Renewable Energy Sector.” In: *2021 IEEE 23rd Conference on Business Informatics (CBI)*. 2021 IEEE 23rd Conference on Business Informatics (CBI). Vol. 01. Sept. 2021, pp. 11–20. DOI: 10.1109/CBI52690.2021.00012.
- [29] Mario J. Kofler, Christian Reinisch, and Wolfgang Kastner. “A Semantic Representation of Energy-Related Information in Future Smart Homes.” In: *Energy and Buildings* 47 (Apr. 2012), pp. 169–179. ISSN: 03787788. DOI: 10.1016/j.enbuild.2011.11.044. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0378778811005901> (visited on 11/03/2022).
- [30] Joanna Kott and Marek Kott. “Generic Ontology of Energy Consumption Households.” In: *Energies* 12.19 (19 Jan. 2019), p. 3712. ISSN: 1996-1073. DOI: 10.3390/en12193712. URL: <https://www.mdpi.com/1996-1073/12/19/3712> (visited on 12/07/2022).
- [31] Serge P. Kovalyov and Olga V. Lukinova. “Integrated Heat and Electric Energy Ontology for Digital Twins of Active Distribution Grids.” In: *AIP Conference Proceedings* 2552.1 (Jan. 5, 2023), p. 080005. ISSN: 0094-243X. DOI: 10.1063/5.0111541. URL: <https://aip.scitation.org/doi/abs/10.1063/5.0111541> (visited on 01/14/2023).
- [32] Dilek Küçük and Doğan Küçük. *OntoWind: An Improved and Extended Wind Energy Ontology*. Mar. 7, 2018. arXiv: 1803.02808 [cs]. URL: <http://arxiv.org/abs/1803.02808> (visited on 11/03/2022).
- [33] Dilek Küçük et al. “PQONT: A Domain Ontology for Electrical Power Quality.” In: *Advanced Engineering Informatics* 24.1 (Jan. 2010), pp. 84–95. ISSN: 14740346. DOI: 10.1016/j.aei.2009.06.009. URL: <https://linkinghub.elsevier.com/retrieve/pii/S1474034609000317> (visited on 11/03/2022).
- [34] Maxime Lefrançois. “Planned ETSI SAREF Extensions Based on the W3C&OGC SOSA/SSN-compatible SEAS Ontology Paaerns.” In: (2017), p. 9.
- [35] Yehong Li et al. “Enhancing Energy Management at District and Building Levels via an EM-KPI Ontology.” In: *Automation in Construction* 99 (Mar. 2019), pp. 152–167. ISSN: 09265805. DOI: 10.1016/j.autcon.2018.12.010. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0926580517309494> (visited on 11/03/2022).
- [36] Phillip Lord and Jennifer Warrender. “Horned-OWL: Building Ontologies at Big Data Scale.” In: ().

- [37] Clement Lork et al. “An Ontology-Based Framework for Building Energy Management with IoT.” In: *Electronics* 8.5 (Apr. 30, 2019), p. 485. ISSN: 2079-9292. DOI: 10.3390/electronics8050485. URL: <https://www.mdpi.com/2079-9292/8/5/485> (visited on 12/01/2022).
- [38] Ravi Lourdasamy and Antony John. “A Review on Metrics for Ontology Evaluation.” In: *2018 2nd International Conference on Inventive Systems and Control (ICISC)*. 2018 2nd International Conference on Inventive Systems and Control (ICISC). Coimbatore: IEEE, Jan. 2018, pp. 1415–1421. ISBN: 978-1-5386-0807-4. DOI: 10.1109/ICISC.2018.8399041. URL: <https://ieeexplore.ieee.org/document/8399041/> (visited on 05/01/2023).
- [39] Filippas Lygerakis, Nikos Kampelis, and Dionysia Kolokotsa. “Knowledge Graphs’ Ontologies and Applications for Energy Efficiency in Buildings: A Review.” In: *Energies* 15.20 (20 Jan. 2022), p. 7520. ISSN: 1996-1073. DOI: 10.3390/en15207520. URL: <https://www.mdpi.com/1996-1073/15/20/7520> (visited on 01/17/2023).
- [40] Jan Morbach, Andreas Wiesner, and Wolfgang Marquardt. “OntoCAPE—A (Re)Usable Ontology for Computer-Aided Process Engineering.” In: *Computers & Chemical Engineering*. Selected Papers from the 18th European Symposium on Computer Aided Process Engineering (ESCAPE-18) 33.10 (Oct. 14, 2009), pp. 1546–1556. ISSN: 0098-1354. DOI: 10.1016/j.compchemeng.2009.01.019. URL: <https://www.sciencedirect.com/science/article/pii/S0098135409000362> (visited on 03/06/2023).
- [41] Fedor S. Nepsha et al. “Development of an Ontology for Smart Distributed Energy Systems \*.” In: *IFAC PapersOnLine*. 11th IFAC Symposium on Control of Power and Energy Systems CPES 2022 55.9 (Jan. 1, 2022), pp. 454–459. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2022.07.079. URL: <https://www.sciencedirect.com/science/article/pii/S2405896322004645> (visited on 01/17/2023).
- [42] Astrid Nieße et al. “Nfdi4energy – National Research Data Infrastructure for the Interdisciplinary Energy System Research.” In: (July 6, 2022). DOI: 10.5281/zenodo.6772013. URL: <https://zenodo.org/record/6772013> (visited on 05/08/2023).
- [43] *Notation3 (N3): A Readable RDF Syntax*. URL: <https://www.w3.org/TeamSubmission/n3/> (visited on 12/13/2022).
- [44] *OpenADR Ontology: Semantic Enrichment of Demand Response Strategies in Smart Grids | IEEE Conference Publication | IEEE Xplore*. URL: <https://ieeexplore.ieee.org/document/9203093> (visited on 12/21/2022).
- [45] Leif Oppermann et al. “Finding and Analysing Energy Research Funding Data: The EnArgus System.” In: *Energy and AI* 5 (Sept. 2021), p. 100070. ISSN: 26665468. DOI: 10.1016/j.egyai.2021.100070. URL: <https://linkinghub.elsevier.com/retrieve/pii/S2666546821000240> (visited on 11/09/2022).
- [46] *OWL 2 Web Ontology Language Document Overview (Second Edition)*. URL: [https://www.w3.org/TR/2012/REC-owl2-overview-20121211/#Relationship\\_to\\_OWL\\_1](https://www.w3.org/TR/2012/REC-owl2-overview-20121211/#Relationship_to_OWL_1) (visited on 12/21/2022).
- [47] *OWL 2 Web Ontology Language New Features and Rationale (Second Edition)*. URL: [https://www.w3.org/TR/2012/REC-owl2-new-features-20121211/#F4:\\_Self\\_Restriction](https://www.w3.org/TR/2012/REC-owl2-new-features-20121211/#F4:_Self_Restriction) (visited on 12/21/2022).
- [48] *OWL 2 Web Ontology Language New Features and Rationale (Second Edition)*. URL: [https://www.w3.org/TR/2012/REC-owl2-new-features-20121211/#F4:\\_Self\\_Restriction](https://www.w3.org/TR/2012/REC-owl2-new-features-20121211/#F4:_Self_Restriction) (visited on 12/21/2022).
- [49] *OWL 2 Web Ontology Language Primer (Second Edition)*. URL: <https://www.w3.org/TR/owl2-primer/> (visited on 12/13/2022).
- [50] *OWL 2 Web Ontology Language RDF-Based Semantics (Second Edition)*. URL: <https://www.w3.org/TR/owl2-rdf-based-semantics/> (visited on 01/10/2023).
- [51] *OWL Web Ontology Language Guide*. URL: <https://www.w3.org/TR/2004/REC-owl-guide-20040210/> (visited on 12/17/2022).
- [52] *OWL Web Ontology Language Semantics and Abstract Syntax*. URL: <https://www.w3.org/TR/owl-semantics/> (visited on 12/17/2022).
- [53] María Poveda-Villalón. “A Reuse-Based Lightweight Method for Developing Linked Data Ontologies and Vocabularies.” In: *The Semantic Web: Research and Applications*. Ed. by Elena Simperl et al. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2012, pp. 833–837. ISBN: 978-3-642-30284-8. DOI: 10.1007/978-3-642-30284-8\_66.
- [54] María Poveda-Villalón, Asunción Gómez-Pérez, and Mari Carmen Suárez-Figueroa. “OOPS! (Ontology Pitfall Scanner!): An On-line Tool for Ontology Evaluation.” In: *International Journal on Semantic Web and Information Systems (IJSWIS)* 10.2 (2014), pp. 7–34. ISSN: 1552-6283. DOI: 10.4018/ijswis.2014040102. URL: <https://www.igi-global.com/article/oops-ontology-pitfall-scanner/116450> (visited on 12/21/2022).
- [55] Marco Pritoni et al. “Metadata Schemas and Ontologies for Building Energy Applications: A Critical Review and Use Case Analysis.” In: *Energies* 14.7 (7 Jan. 2021), p. 2024. ISSN: 1996-1073. DOI: 10.3390/en14072024. URL: <https://www.mdpi.com/1996-1073/14/7/2024> (visited on 01/17/2023).

- [56] *RDF 1.1 N-Triples*. URL: <https://www.w3.org/TR/n-triples/> (visited on 12/13/2022).
- [57] *RDF 1.1 Primer*. URL: <https://www.w3.org/TR/rdf11-primer/> (visited on 12/13/2022).
- [58] *RDF 1.1 Turtle*. URL: <https://www.w3.org/TR/turtle/> (visited on 12/13/2022).
- [59] *RDF Schema 1.1*. URL: <https://www.w3.org/TR/rdf-schema/> (visited on 12/13/2022).
- [60] Christian Reinisch et al. “ThinkHome Energy Efficiency in Future Smart Homes.” In: *EURASIP Journal on Embedded Systems* 2011.1 (2011), p. 104617. ISSN: 1687-3963. DOI: 10.1155/2011/104617. URL: <http://jes.erasipjournals.com/content/2011/1/104617> (visited on 01/19/2023).
- [61] Jenn Riley. “UNDERSTANDING METADATA.” In: ().
- [62] Thomas Rosnet et al. “FAIR-Checker, a Web Tool to Support the Findability and Reusability of Digital Life Science Resources.” July 6, 2021. DOI: 10.5281/zenodo.5914307. URL: <https://zenodo.org/record/5914307> (visited on 05/04/2023).
- [63] Djamel Saba et al. “Development of an Ontology-Based Generic Optimisation Tool for the Design of Hybrid Energy Systems.” In: (), p. 12.
- [64] Gabriel Santos et al. “Electricity Markets Ontology to Support MASCEM’s Simulations.” In: *Highlights of Practical Applications of Scalable Multi-Agent Systems. The PAAMS Collection*. Ed. by Javier Bajo et al. Vol. 616. Communications in Computer and Information Science. Cham: Springer International Publishing, 2016, pp. 393–404. ISBN: 978-3-319-39386-5 978-3-319-39387-2. DOI: 10.1007/978-3-319-39387-2\_33. URL: [http://link.springer.com/10.1007/978-3-319-39387-2\\_33](http://link.springer.com/10.1007/978-3-319-39387-2_33) (visited on 12/07/2022).
- [65] Nazaraf Shah et al. “Ontology for Home Energy Management Domain.” In: *Digital Information and Communication Technology and Its Applications*. Ed. by Hocine Cherifi, Jasni Mohamad Zain, and Eyas El-Qawasmeh. Vol. 167. Communications in Computer and Information Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 337–347. ISBN: 978-3-642-22026-5 978-3-642-22027-2. DOI: 10.1007/978-3-642-22027-2\_28. URL: [http://link.springer.com/10.1007/978-3-642-22027-2\\_28](http://link.springer.com/10.1007/978-3-642-22027-2_28) (visited on 12/06/2022).
- [66] *Standards*. xkcd. URL: <https://xkcd.com/927/> (visited on 12/14/2022).
- [67] Thanos G. Stavropoulos et al. “BOnSAI: A Smart Building Ontology for Ambient Intelligence.” In: *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics - WIMS '12*. The 2nd International Conference. Craiova, Romania: ACM Press, 2012, p. 1. ISBN: 978-1-4503-0915-8. DOI: 10.1145/2254129.2254166. URL: <http://dl.acm.org/citation.cfm?doid=2254129.2254166> (visited on 11/03/2022).
- [68] Mari Carmen Suárez-Figueroa. “NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse.” PhD thesis. Universidad Politécnica de Madrid, June 25, 2010. DOI: 10.20868/UPM.thesis.3879. URL: <http://oa.upm.es/3879/> (visited on 11/03/2022).
- [69] Nikola M. Tomašević et al. “Ontology-Based Facility Data Model for Energy Management.” In: *Advanced Engineering Informatics*. Collective Intelligence Modeling, Analysis, and Synthesis for Innovative Engineering Decision Making 29.4 (Oct. 1, 2015), pp. 971–984. ISSN: 1474-0346. DOI: 10.1016/j.aei.2015.09.003. URL: <https://www.sciencedirect.com/science/article/pii/S1474034615000981> (visited on 01/17/2023).
- [70] *W3C Web Ontology (WebOnt) Working Group (OWL) (Closed)*. URL: <https://www.w3.org/2001/sw/WebOnt/> (visited on 11/09/2022).
- [71] August Wierling et al. “FAIR Metadata Standards for Low Carbon Energy Research—A Review of Practices and How to Advance.” In: *Energies* 14.20 (Oct. 15, 2021), p. 6692. ISSN: 1996-1073. DOI: 10.3390/en14206692. URL: <https://www.mdpi.com/1996-1073/14/20/6692> (visited on 11/03/2022).
- [72] Mark D. Wilkinson et al. “The FAIR Guiding Principles for Scientific Data Management and Stewardship.” In: *Scientific Data* 3.1 (Dec. 2016), p. 160018. ISSN: 2052-4463. DOI: 10.1038/sdata.2016.18. URL: <http://www.nature.com/articles/sdata201618> (visited on 11/03/2022).
- [73] Sheng-Yuan Yang. “Developing an Energy-Saving and Case-Based Reasoning Information Agent with Web Service and Ontology Techniques.” In: *Expert Systems with Applications* 40.9 (July 2013), pp. 3351–3369. ISSN: 09574174. DOI: 10.1016/j.eswa.2012.12.044. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0957417412012857> (visited on 11/03/2022).
- [74] Marcia Lei Zeng and Jian Qin. *Metadata, Second Edition*. 2nd ed. Atlanta, GA, USA: ALA Neal-Schuman, 2016. 584 pp. ISBN: 978-1-55570-965-5.



## B. List of Figures

Figure 1.	Code snippet formatting veterinarian contact information in HTML (left) and XML (right). . . . .	5
Figure 2.	The example RDF statement <code>works_at(veterinarian, animal_clinic)</code> in graph form	6
Figure 3.	Code snippet formatting the graph from Figure 2 in RDF using XML. . . . .	6
Figure 4.	Code Snippets restricting the works at relation from Figure 2 in RDFS using XML. Full code can be found in Appendix A . . . . .	7
Figure 5.	The relationship between the standard OWL1 sublanguages visualized. . . . .	7
Figure 6.	Code snippet showing the first lines of a OWL1 ontology. . . . .	8
Figure 7.	RDFS/OWL code snippet for showcasing classes given in subsection 2.3.1 for an OWL2 ontology. . . . .	8
Figure 8.	RDFS/OWL code snippet for showcasing unique individuals given in subsection 2.3.1 for an OWL2 ontology. The use of <code>owl:distinctMembers</code> makes the members in the collection unique. . . . .	9
Figure 9.	The relationship between the standard OWL2 sublanguages visualized. . . . .	10
Figure 10.	Modified Prisma approach used for ontology procurement . . . . .	15
Figure 11.	Spiderweb diagrams outlining the score in each category for each of the ontologies.	32
Figure 11a.	Spiderweb diagram outlining the score in each category for each of the bottom four ontologies. . . . .	32
Figure 11b.	Spiderweb diagram outlining the score in each category for each of the top four ontologies. . . . .	32
Figure 12.	TIB Terminology Service Homepage. Screenshot taken 03.05.2023 . . . . .	34
Figure 13.	TIB Terminology Service search page. Screenshot taken 03.05.2023 . . . . .	34
Figure 14.	Sketch showing the functionality of the planned tool. . . . .	35
Figure 15.	OLS4 homepage after integration. . . . .	36
Figure 15a.	OLS4 Homepage with added refined search. Search bar closed. . . . .	36
Figure 15b.	OLS4 homepage with added refined search. Search bar open. . . . .	36
Figure 16.	Modified OLS4 search results page. . . . .	36
Figure 17.	Full RDFS code for the example given in subsection 2.3.1. . . . .	41
Figure 18.	First part of the RDFS/OWL code for the example given in subsection 2.3.1. . . . .	42
Figure 19.	Second part of the RDFS/OWL code for the example given in subsection 2.3.1. . . . .	43
Figure 20.	Third part of the RDFS/OWL code for the example given in subsection 2.3.1. . . . .	44



## C. List of Tables

Table 1.	A table comparing OWL1 sublanguages. . . . .	7
Table 2.	A table comparing OWL2 sublanguages. . . . .	9
Table 3.	A table listing relevant ontology editors. . . . .	11
Table 4.	A table listing some reasoners. . . . .	12
Table 5.	Ontologies identified after second screening. . . . .	16
Table 6.	Ontologies identified for comparison. . . . .	18
Table 7.	Ontology comparison categories. . . . .	22
Table 8.	Best Practice Comparison. . . . .	26
Table 9.	Practical Implementation Comparison. . . . .	27
Table 10.	Maintenance and Accessibility Comparison . . . . .	28
Table 11.	Governance Comparison . . . . .	30
Table 12.	Accumulated scores for each ontology. . . . .	31
Table 13.	Ontologies used by each ontology of the comparison. . . . .	45
Table 14.	Classes and properties of the compared ontologies. . . . .	48
Table 15.	Found Sourcecode for each ontology. . . . .	49



## D. Declaration of Academic Integrity

I hereby declare that I have composed the presented thesis independently, on my own, and without any other resources than the ones indicated. All thoughts taken directly or indirectly from external sources are properly denoted as such. This thesis has neither been previously submitted to another authority nor has it been published yet.

Ahnsbeck, May 22, 2023

(Place, Date)

A handwritten signature in blue ink that reads "A. Steiner". The signature is written in a cursive style with a large initial 'A' and a long, sweeping underline.

(Signature)