RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN
INSTITUT FÜR INFORMATIK III

MASTER THESIS

Survey on Metadata Repositories for Vocabularies and Ontologies

Ana Cristina TRILLOS UJUETA
E-mail: trillos@informatik.uni-bonn.de

1. Examiner: Prof. Dr. Jens LEHMANN
2. Examiner: Dr. Maria-Esther VIDAL
Supervisors: Dr. Ioanna LYTRA - Dr. Fabrizio ORLANDI

A thesis submitted in fulfillment of the requirements for the degree of Master of Science

in the

Smart Data Analytics
Institute of Computer Science

June 1, 2018

Declaration of Authorship

I, Ana Cristina TRILLOS UJUETA, declare that this thesis titled, “Survey on Metadata Repositories for Vocabularies and Ontologies” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN INSTITUT FÜR INFORMATIK III

Abstract

Faculty of Mathematics and Natural Science
Institute of Computer Science

Master of Science

Survey on Metadata Repositories for Vocabularies and Ontologies

by Ana Cristina TRILLOS UJUETA

With the increase of Semantic Web and Linked Open Data documents such as vocabularies and ontologies, there is a need to group these documents in a commonplace/system where all kinds of users, experts, and non-experts, are able to find, reuse and share their ontologies and vocabularies with the world. This kind of system is called Metadata Repository. Currently, there exist several of them that are public or private but the majority of them are not domain specific. By performing a literature survey, it was studied which repositories already exist in different domains and their requirements and functionalities. From a case study in the context of the Big Data Ocean EU funded project, a methodology was determined to help decide when an already existing metadata repository can be extended and reused in another project, or if it is better to develop a new system.

The methodology proposed is the union of the features and the data model found in the studied metadata repositories plus the requirements of the BigDataOcean Metadata Repository. The BigDataOcean Metadata Repository was evaluated by users through two studies, a task scenario test followed by a usability test to determine the usability, what difficulties the users encountered and what users think about the system.

As a result of the literature survey, the lack of documentation of some repositories made difficult or impossible to evaluate their functionalities and data model. From the BigDataOcean Metadata Repository user evaluation, it is possible to conclude that the system is easy to use by experts and non-experts users, and it does not require support from a technical person in order to use all the features.

Acknowledgements

I would like to thank Dr. Ioanna Lytra and Dr. Fabrizio Orlandi for their support and guidance through the development of the BigDataOcean Metadata Repository tool and this master thesis. I would like also to thank all participants of the user evaluation.

Contents

List of Figures

List of Tables

Dedicated to my parents and brother…

Chapter 1

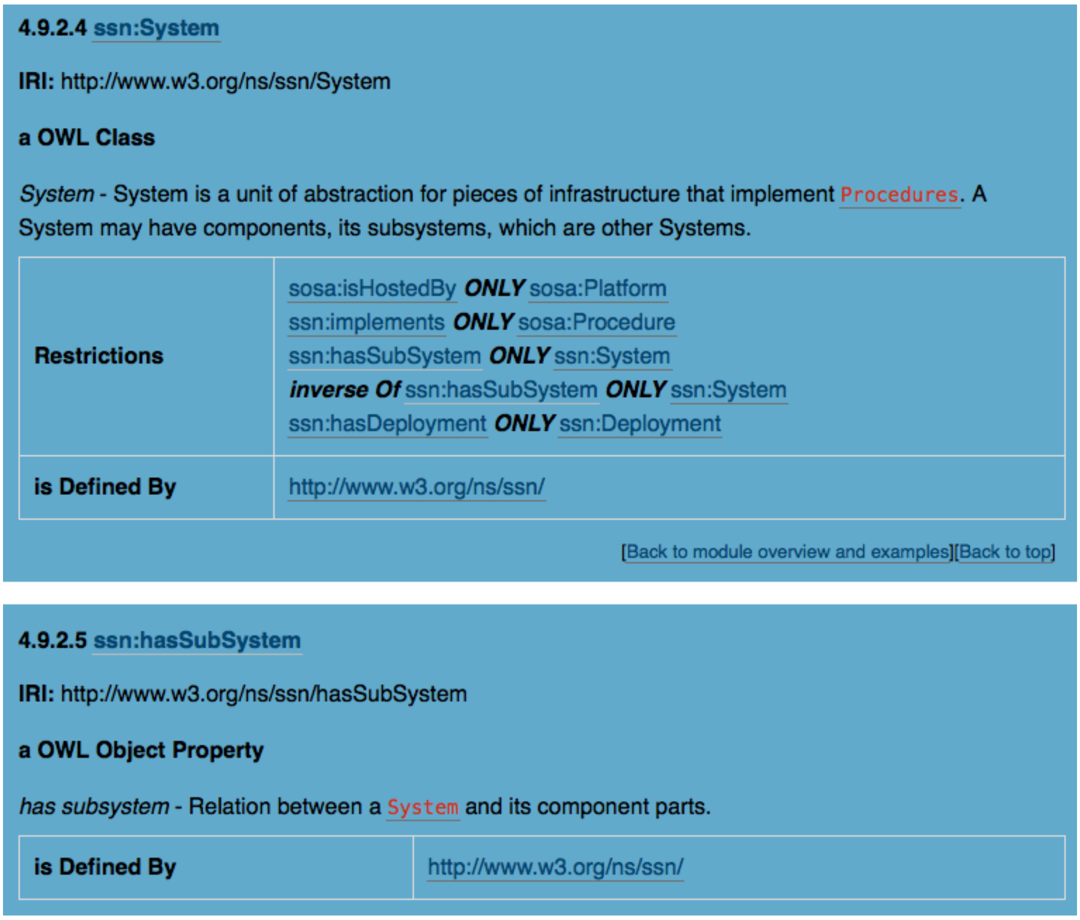
Introduction

Semantic Web is the Web of Linked Data [1]. This data is linked using vocabularies and ontologies, and described with standard formats, like RDF[1] (Resource Description Format) or OWL[2] (Web Ontology Language). Having linked data allows Semantic Web technologies to manage, query and make inferences about the data. These data should be reached and shared Online [2], with a unique identifier (URI). The URI allows users and Semantic Web experts to reuse the data, and it needs to be Linked Open Data (LOD). To have LOD, a five-star rating system was developed[3]; where it defines, LOD needs to have an open license, be machine-readable, to be described in a non-proprietary format, be following RDF Standards and finally to be linked to another documents or data.

LOD documents such as Vocabularies and Ontologies define terminology used for characterizing data from one or more domains. As mentioned in [1] there is no apparent difference between vocabulary and ontology, but the tendency is to refer to ontologies to more complex and formal semantic web documents.
In this thesis, the usage of the terms ontology and vocabulary is interchangeable, without defining any difference between them.

Since the introduction of the Semantic Web, there has been an increasing amount of vocabularies and ontologies describing data in specific domains, e.g., Semantic Sensor Network (SSN)[4] for defining terminology used to characterize sensors in the maritime domain. Figure 1.1 shows two terms defined by SSN, `ssn:System` is a class used to characterize a "unit of abstraction for pieces of infrastructure that implement Procedures," and `ssn:hasSubSystem` is a property for specifying the relation between a system and its components. But also for all domains, e.g., Dublin Core Metadata Initiative, Metadata Terms vocabulary[5]. In Figure 1.2 two properties are shown, `dct:created` and `dct:creator` for defining the creation date and the creator of a resource, respectively.

[1] https://www.w3.org/RDF/
[2] https://www.w3.org/OWL/
[3] https://www.w3.org/DesignIssues/LinkedData.html
[4] https://www.w3.org/TR/vocab-ssn/
[5] http://dublincore.org/documents/dcmi-terms/


4.9.2.4 ssn:System

IRI: http://www.w3.org/ns/ssn/System

a OWL Class

System - System is a unit of abstraction for pieces of infrastructure that implement Procedures. A System may have components, its subsystems, which are other Systems.

Restrictions	sosa:isHostedBy ***ONLY*** sosa:Platform ssn:implements ***ONLY*** sosa:Procedure ssn:hasSubSystem ***ONLY*** ssn:System ***inverse Of*** ssn:hasSubSystem ***ONLY*** ssn:System ssn:hasDeployment ***ONLY*** ssn:Deployment
is Defined By	http://www.w3.org/ns/ssn/

[Back to module overview and examples][Back to top]

4.9.2.5 ssn:hasSubSystem

IRI: http://www.w3.org/ns/ssn/hasSubSystem

a OWL Object Property

has subsystem - Relation between a System and its component parts.

is Defined By	http://www.w3.org/ns/ssn/

FIGURE 1.1: Example of a class and a property from Semantic Sensor Network Ontology.

This increase makes harder for researchers and users to search and share relevant and standardized ontologies and vocabularies. To be able to determine whether an ontology or vocabulary is appropriate or it fulfills the researcher/user needs, users require a place to store, search and share metadata about the vocabularies and ontologies. Something like a big database storing all information about a vocabulary or ontology, accessible to users from anywhere in the world.

This database-like is called Metadata Repository. A repository is a database for storing, managing and querying data. The name, creation date, description of a resource are metadata aspects; information that describes resources, or in this case specifically, describes vocabularies and ontologies. The metadata stored about a vocabulary or ontology should be detailed information than just format and name. A metadata repository needs to be accessible, extensible and expressive.

Currently, there exist metadata repositories which are domain-specific, e.g., BioPortal [2] in the biomedical domain or the MMI Ontology Registry and Repository (MMI-ORR) [3] for the maritime domain; or multi-domain repositories, such as Ontohub [4], and Linked Open Vocabularies (LOV) [5],

Term Name:	**created**
URI:	http://purl.org/dc/terms/created
Label:	Date Created
Definition:	Date of creation of the resource.
Type of Term:	Property
Refines:	http://purl.org/dc/elements/1.1/date
Refines:	http://purl.org/dc/terms/date
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal
Version:	http://dublincore.org/usage/terms/history/#created-003
Term Name:	**creator**
URI:	http://purl.org/dc/terms/creator
Label:	Creator
Definition:	An entity primarily responsible for making the resource.
Comment:	Examples of a Creator include a person, an organization, or a service.
Type of Term:	Property
Refines:	http://purl.org/dc/elements/1.1/creator
Refines:	http://purl.org/dc/terms/contributor
Has Range:	http://purl.org/dc/terms/Agent
Version:	http://dublincore.org/usage/terms/history/#creatorT-002
EquivalentProperty:	http://xmlns.com/foaf/0.1/maker

FIGURE 1.2: Example of properties from DCMI Metadata Terms.

among others. These repositories have one purpose in common, to share and facilitate search and selection of the relevant and adequate vocabulary and ontology for any task according to the metadata.

This thesis gives an overview of existing metadata repositories in different domains, their features/functionalities and how they are being used. It presents a use case in the context of the BigDataOcean EU funded project for building a Metadata Repository in order to propose a methodology on how to create or extend an existing metadata repository based on the requirements for a specific repository.

The thesis is structured as follows: Chapter 1 offers an introduction. In Chapter 2 the literature survey evaluates features, interfaces and data model of existing Metadata Repositories. Chapter 3 has two sections; the first one presents the case study "BigDataOcean Metadata Repository" and the second introduces a methodology for creating a metadata repository from experience gained in the case study and the literature survey results. Chapter 4 presents the user evaluation of BigDataOcean Metadata Repository case study. Finally, Chapter 5 and 6 offer discussions, conclusions, and future work.

Chapter 2

Literature Survey

In this chapter, the methodology implemented to search and select metadata repositories' documentation suitable for this study is described. Later, a survey of the chosen papers and systems divided into three categories, (1) features and functionalities, (2) interfaces and (3) data model.

2.1 Methodology

2.1.1 Collection

For searching and collecting publications, it was used two digital libraries, Google Scholar[1] and ACM Digital Library[2].
ACM is a complete digital library that offers advanced query execution, the ability to navigate through references and citations. On the other hand, Google Scholar is free and allows to navigate through citations.

It was executed advanced queries with the words *"metadata repository"*, *"metadata registry"*, *"Link Open Data"*, and *"semantic web"*, Figure 2.1 shows three query examples performed over the digital libraries.
This search resulted in 27 publications related to the topic Metadata repository.

2.1.2 Inclusion and Exclusion Criteria

It was necessary to determine inclusion/exclusion criteria for selecting the more suitable publications and systems. These criteria are:

1. The repository must have at least one publication.
2. The publication must focus only on the repository features and design.
3. The repository should be mentioned in others repository's publications.
4. The repository must be related to Semantic Web.

[1] `https://scholar.google.de`
[2] `https://dl.acm.org`

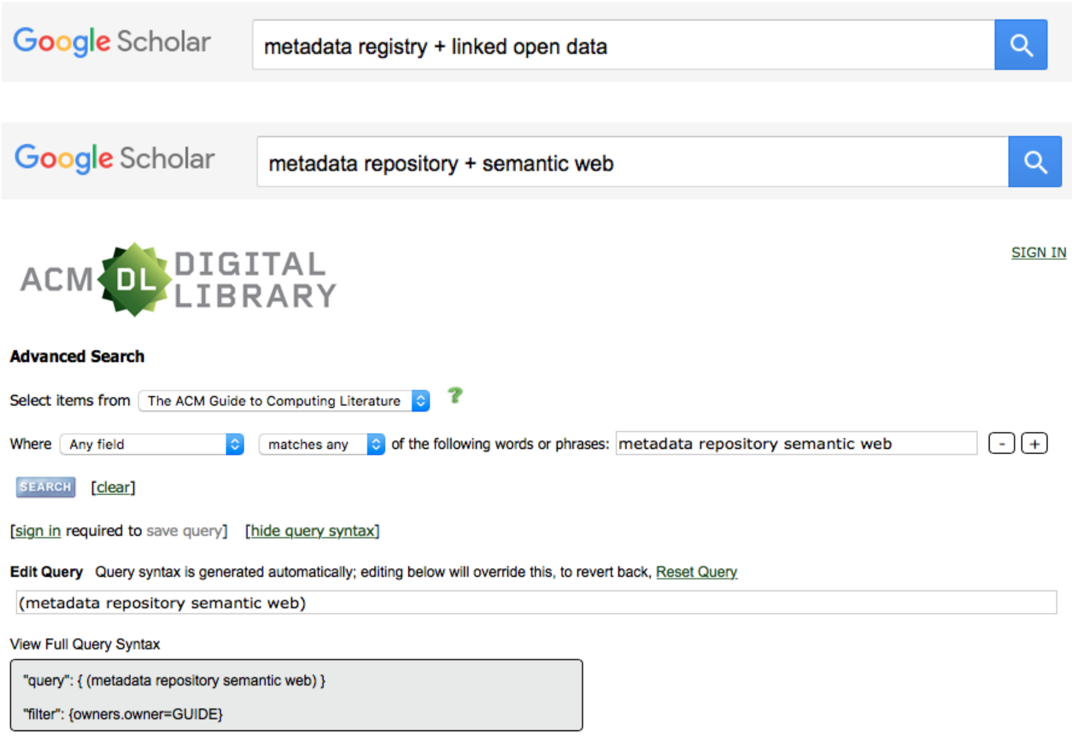


FIGURE 2.1: Search Queries for ACM Digital Library and Google Scholar

From those 27 publications, the following 8 are included in the final review.
OUIP, Ontology-based UMLS Integration Project [6] is a metadata management system for medical domains.
Swoogle [7] [8] is a multi-domain search and metadata engine for Semantic Web.
Oyster [9] is a Peer-to-Peer ontology sharing system.
BioPortal [2] [10] is a system for sharing biomedical ontologies.
Cupboard [11] is an ontology publishing, sharing and reuse system, that uses WATSON engine for ontology indexing. It relies on the Oyster system, which provides the ability to record metadata about the ontology.
MMI-ORR, Marine Metadata Initiative Ontology Registry and Repository [3] [12] is a metadata repository for marine ontologies.
OntoHub [4] [13] is an engine that manages distributed heterogeneous ontologies.
LOV, Linked Open Vocabularies [5] [14] is a catalog for reusable multi-domain ontologies and vocabularies.

Table 2.1 shows which criteria are accomplished and which not by each of the eight final publications. Swoogle, Oyster, BioPortal, and OntoHub achieved all four inclusion/exclusion criteria. Cupboard, MMI-ORR, and LOV accomplished tree of four inclusion/exclusion criteria. OUIP achieved only two of four criteria points, and was studied because it was one of the first metadata repositories found in the collection process.

Repositories	C.1	C.2	C.3	C.4
OUIP	☺	☺	☹	☹
Swoogle	☺	☺	☺	☺
Oyster	☺	☺	☺	☺
BioPortal	☺	☺	☺	☺
Cupboard	☺	☺	☺	☹
MMI-ORR	☺	☺	☹	☺
OntoHub	☺	☺	☺	☺
LOV	☺	☺	☹	☺

TABLE 2.1: List of repositories after inclusion/exclusion criteria. ☺accomplish ☹not accomplish

Among the excluded publications, it is possible to find one paper that met three of four inclusion/exclusion criteria points. **Ontoselect** [15] has at least one publication, it was mentioned in other repository's publication and it is related to Semantic Web, but it was not selected to be evaluated because the paper does not indicate all the features and their implementations aspects. Ontoselect is not available online which makes impossible to assess the system objectively.
HealthFinland [16] and **MuseumFinland** [17] were also excluded because both systems have at least one publication and are related to Semantic Web, but these systems are not mentioned in other publications found, and the respective publication does not focus on the implementation of the system's features.

2.2 Results

The goal of the literature survey is to evaluate the features, interfaces and data model of the repositories, this will help to determine the essential characteristics of a metadata repository, and which vocabularies and ontologies are useful for accurately describing vocabularies, ontologies, and datasets.
Figure 2.2 shows two of the evaluation categories, how they are divided and which interfaces require which feature in order to work. For example, the interface "Display Metadata" needs the features, CRUD functionality, Search and Links. Later on, why this occurs will be explained.

[3] http://swoogle.umbc.edu/2006/
[4] https://sourceforge.net/projects/oyster2/
[5] https://bioportal.bioontology.org
[6] https://github.com/ncbo
[7] http://mmisw.org/ont/#
[8] https://github.com/mmisw/orr-ont
[9] https://github.com/mmisw/orr-portal
[10] https://ontohub.org
[11] https://github.com/ontohub/ontohub
[12] http://lov.okfn.org/dataset/lov/

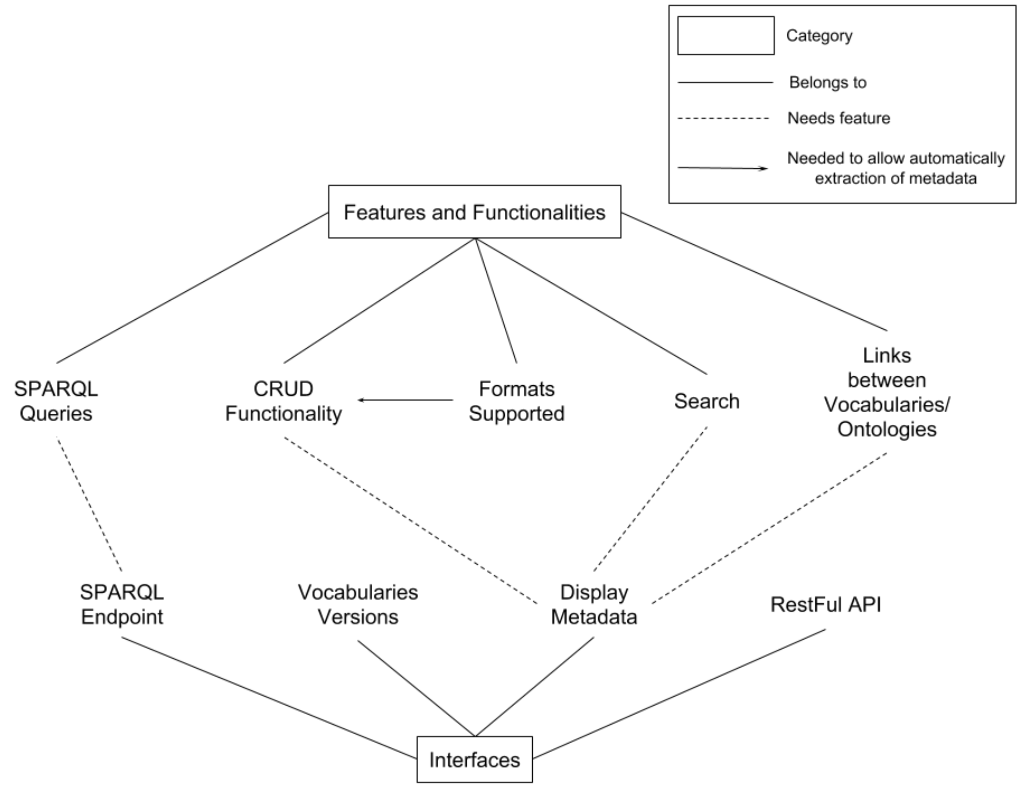


FIGURE 2.2: The taxonomy used in the Literature Survey Evaluation.

Table 2.2 presents a general comparison of repositories, online availability, and whether they are open source or not, and where to find their source code. As shown in the table, only two repositories were not reachable online, OUIP and Cupboard, the rest of them are online available. The majority of repositories are multi-domain, and there is a low amount of domain-specific repositories, for example, OUIP and BioPortal's domain is (bio)medical, and the MMI-ORR's domain is marine. Four of the eight evaluated repositories, BioPortal, MMI-ORR, Ontohub, and LOV, have published online their source code.

2.2.1 Features of Metadata Repositories

As mentioned on [18], providing an access point to online ontologies and semantic data is the goal of a Semantic Web gateway. Thus, this is obtained by having features of a traditional search engine; data collection and analysis, and being able to execute queries, adapted to the Semantic Web.

Table 2.3 is a summary of the features the metadata repositories implement or not, the following subsections present an explanation of each feature and how each repository implements each feature.

[13]https://github.com/pyvandenbussche/lov
[14]https://github.com/pyvandenbussche/lovScripts

Repository	Domain	Reachable Online?	Source Code	License
OUIP	Medical	No	N/A	N/A
Swoogle	Multi-Domain	Yes[3]	N/A	Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License
Oyster	Multi-Domain	Yes[4]	N/A	N/A
BioPortal	Biomedical	Yes[5]	Yes[6]	N/A
Cupboard	Multi-Domain	No	N/A	N/A
MMI-ORR	Marine	Yes[7]	Yes[8 9]	N/A
OntoHub	Multi-Domain	Yes[10]	Yes[11]	GNU AGPL 3.0 license
LOV	Multi-Domain	Yes[12]	Yes[13 14]	Creative Commons Attribution 4.0 International License.

TABLE 2.2: Domain, online availability, online source-code and license comparison of the final eight metadata repositories.

Repository	CRUD	SPARQL Queries	Search	Formats Supported	Visualizations
OUIP	Yes	Yes	Yes	N/A	Yes
Swoogle	No	No	Yes	R^+, NT, N3	Yes
Oyster	Yes	No	Yes	O, D, R*	No
BioPortal	Yes	No	Yes	O, R, O*	No
Cupboard	Yes	N/A	N/A	N/A	N/A
MMI-ORR	Yes	Yes	Yes	R, O	N/A
OntoHub	Yes	No	N/A	O, R, CL, FL, O*, X	N/A
LOV	Yes	Yes	Yes	R, R*, O	Yes

OWL (O), DAML+OIL (D), RDF-S (R*), RDF (R), R^+ (RDF/XML), NT (NTriples), N3 (N3), OBO (O*), Common Logic (CL), First-order logic (FL), XML (X)

TABLE 2.3: Identification of which repositories implement which of the following functionalities, CRUD operations, ability to execute SPARQL queries, search vocabularies/ontologies given keywords, which vocabulary's format does the repository support, and whether the repository displays links between vocabularies/ontologies.

2.2.1.1 Insert, Update and Delete Vocabularies and Ontologies

In order to collect semantic data about ontologies and vocabularies, a user should be able to CRUD[15] the system. These operations allow having up-to-date metadata about vocabularies and ontologies.

As shown in Table 2.3 most of the systems evaluated, except for Swoogle, allow to CRUD the system. Usually, to do so, it is required the user has an account and log into the system to create, update or delete metadata.
Swoogle does not allow this feature; instead, it crawls the Internet to automatically discover and generate metadata about the Semantic Web Documents found, and allows users to submit Semantic Web Documents URLs.
Oyster allows the creation of metadata for vocabularies and ontologies, and to import files and automatically extract their metadata when available.
Similar to Oyster, when adding a new vocabulary/ontology in LOV, it extracts metadata from the URI provided by the user allowing the user to

[15]Create, Read, Update and Delete

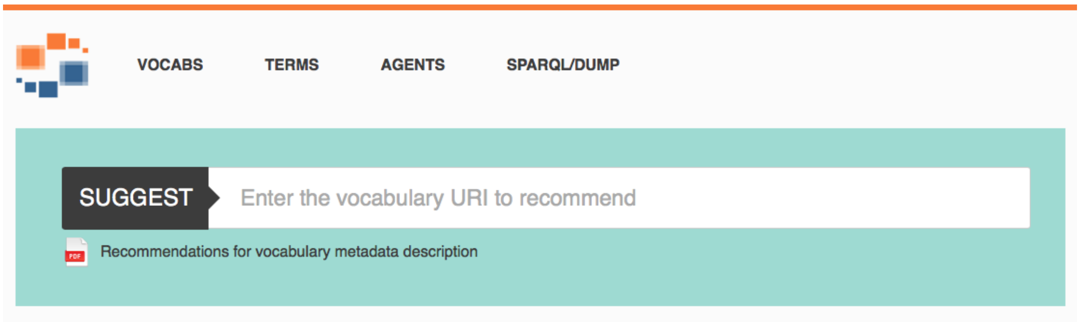


FIGURE 2.3: LOV suggest feature.

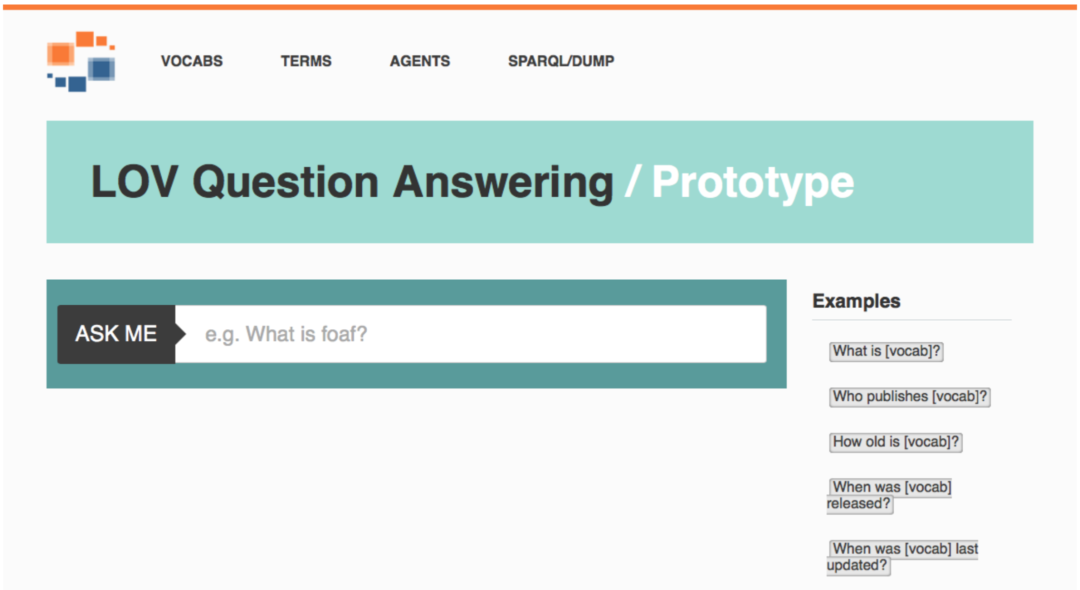


FIGURE 2.4: Q&A section in LOV.

modify metadata before storing it. Another behavior LOV presents, alike to Swoogle's, is the ability to suggest new vocabularies or ontologies via their Suggest button providing the URI of the vocabulary.
The OUIP System [6] claims it lets users manage metadata, add, modify or deactivate code definitions.

2.2.1.2 Perform SPARQL queries

As referred earlier, allowing the execution of SPARQL queries is a technique that lets users perform more specific and rich searches over the metadata repository. This feature requires users to have a certain level of knowledge on SPARQL query rules; or its implementation can be as a Question and Answer (Q&A) system, in which users can ask questions, and after the evaluation of the SPARQL query they receive an answer.

The OUIP System performs SPARQL queries when searching for medical terms. Thus, this system combines two functionalities together, SPARQL querying and searching with keywords. In contrast, MMI-ORR and LOV

repositories offer the ability to perform queries over the repository's semantic graph. Furthermore, LOV offers a Q&A system (Figure 2.4).
Contrary to these repositories, for Swoogle, BioPortal, and Oyster, no evidence exists about whether they offer or not this feature.

2.2.1.3 Search for Vocabularies and Ontologies based on Keywords

Being able to search vocabularies and ontologies based on search criteria or keywords is critical, this supports users while trying to find vocabularies, ontologies and even terms on the metadata repository.

When evaluating Swoogle, there is a discrepancy between the paper [7] and the Online version. The online version shows a single search field, and by selecting between ontology, document or term, it offers the results for the selected option, whereas the paper describes an Advanced Database Search, with more fields for a refined search. This discrepancy could be due to a newer version Online.
Oyster a Peer-to-Peer system, lets users select between performing searches on the Local peer or certain user-selected peers. Ontology searches can be performed using simple keywords or a detail semantic search utilizing Ontology name, language, contributor and more.
Contrary to Oyster, BioPortal offers two different search fields, one for classes and the other for ontologies. When searching for classes, it offers advanced search parameters (Figure 2.11); while searching for ontologies, there is a possibility to search all ontologies or to choose an ontology from a list.
LOV offers the possibility to search vocabularies and ontologies by their prefix, or by words related to their title. It also allows searching for terms, classes, and properties using keywords. The search is performed over Elasticsearch[16], an open-source Java-based search engine based on Lucene.
On the other hand, MMI-ORR allows searches using words that could be in a vocabulary or ontology's title.
OUIP System claims it allows search using keywords for medical terms definitions, and offers different options, e.g., time lime for when terms were activated, and definition libraries for allowing users to search in one or more libraries.

2.2.1.4 Formats Supported

The World Wide Web Consortium (W3C) proposes a wide range of standards, such as RDF and RDFS [19], OWL [20] and SKOS[17] [21] for creating and describing semantic web vocabularies and ontologies; but these standards can be syntactically written into different syntax languages.
Figure 2.5 shows a general example of the structure of RDF/XML [22] syntax. Each node and predicate is enclosed in an XML tag.
In contrast, Figure 2.6 displays the structure of the Turtle [23] syntax. The

[16] https://www.elastic.co/products/elasticsearch
[17] Simple Knowledge Organization System Reference

```
<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
  <ex:editor>
    <rdf:Description>
      <ex:homePage>
        <rdf:Description rdf:about="http://purl.org/net/dajobe/">
        </rdf:Description>
      </ex:homePage>
    </rdf:Description>
  </ex:editor>
</rdf:Description>

<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
  <ex:editor>
    <rdf:Description>
      <ex:fullName>Dave Beckett</ex:fullName>
    </rdf:Description>
  </ex:editor>
</rdf:Description>

<rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
  <dc:title>RDF 1.1 XML Syntax</dc:title>
</rdf:Description>
```

FIGURE 2.5: Example of RDF/XML syntax structure.

Turtle syntax is more human-readable than RDF/XML, and it is written in triples, subject, predicate, and object. Each triple represents a fact about the subject.
Another syntax language is DAML+OIL [24], it is written in RDF/XML-like schema, but with a structured format that is more natural to read, as shown in Figure 2.7.

In this literature survey, "formats supported" refers to those syntactical languages a system is able to understand, in order to automatically extract metadata from an ontology/vocabulary file; it does not refer to how each system is storing the metadata of a vocabulary or ontology. Each system should provide a list of formats supported.

Oyster supports OWL, DAML+OIL[18] and RDFS format files. To facilitate sharing and re-using vocabularies/ontologies, Oyster proposed a metadata standard for characterizing ontologies, called OMV[19].
BioPortal supports documents in OWL, RDF and OBO[20] formats.
Swoogle accepts formats such as, RDF/RDFS, OWL, DAML+OIL and N3[21].
LOV, on the other hand, offers a document describing metadata recommendations a vocabulary should follow to be accepted into the system. It accepts formats RDF, RDFS, and OWL.
Cupboard relies on the Oyster system to allow users describe metadata using OMV format [9].

[18] https://www.w3.org/TR/daml+oil-reference
[19] Ontology Metadata Vocabulary
[20] format of the Protégé frame language
[21] Notation3 - https://www.w3.org/TeamSubmission/n3/

```
@base <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel: <http://www.perceive.net/schemas/relationship/> .

<#green-goblin>
    rel:enemyOf <#spiderman> ;
    a foaf:Person ;    # in the context of the Marvel universe
    foaf:name "Green Goblin" .

<#spiderman>
    rel:enemyOf <#green-goblin> ;
    a foaf:Person ;
    foaf:name "Spiderman", "Человек-паук"@ru .
```

FIGURE 2.6: Example of Turtle syntax structure.

```
<rdf:RDF
  xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:daml="http://www.w3.org/2001/10/daml+oil#"
  xmlns     ="http://www.w3.org/2001/10/daml+oil#"
>

<rdf:Description rdf:about="">
  <versionInfo>$Id: NOTE-daml+oil-reference-20011218.html,v 1.7 2017/10/02 10:23:30 denis Exp $</versionInfo>
  <imports rdf:resource="http://www.w3.org/2000/01/rdf-schema"/>
</rdf:Description>

<!-- (meta) classes of "object" and datatype classes  -->

<rdfs:Class rdf:ID="Class">
  <rdfs:label>Class</rdfs:label>
  <rdfs:comment>
    The class of all "object" classes
  </rdfs:comment>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdfs:Class>

<rdfs:Class rdf:ID="Datatype">
  <rdfs:label>Datatype</rdfs:label>
  <rdfs:comment>
    The class of all datatype classes
  </rdfs:comment>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdfs:Class>
```

FIGURE 2.7: Example of DAML+OIL syntax structure.

The MMI-ORR system allows vocabularies and ontologies represented in various formats; however, RDF and OWL are the most prominent formats. MMI-ORR allows through the component Voc2RDF the creation of vocabularies by importing data from a file written in comma-separated-values (CSV) format.
OntoHub accepts vocabularies and ontologies described in a broader range of formats, among them, OWL, RDF, Common Logic, First-order Logic, OBO, XML.

2.2.1.5 Links between Vocabularies and Ontologies

Displaying incoming and outgoing links between vocabularies and ontologies gives the user a better understanding of how they are used and related to each other.

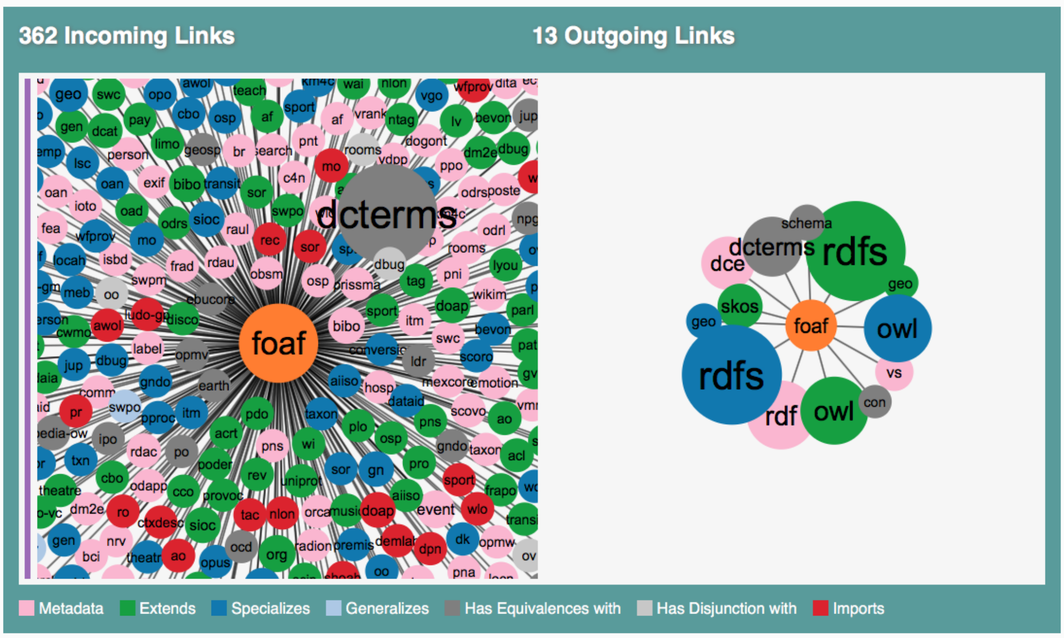


FIGURE 2.8: Example of how LOV shows incoming and outgoing links from FOAF vocabulary.

These vocabularies and ontologies are linked to each other because they reference one or more vocabularies. For example, one vocabulary uses the vocabulary Dublin Core Metadata Element Set [25] properties for describing the title, creator, and description; but at the same time it can use the RDF Concepts Vocabulary [26] for describing new classes or properties.

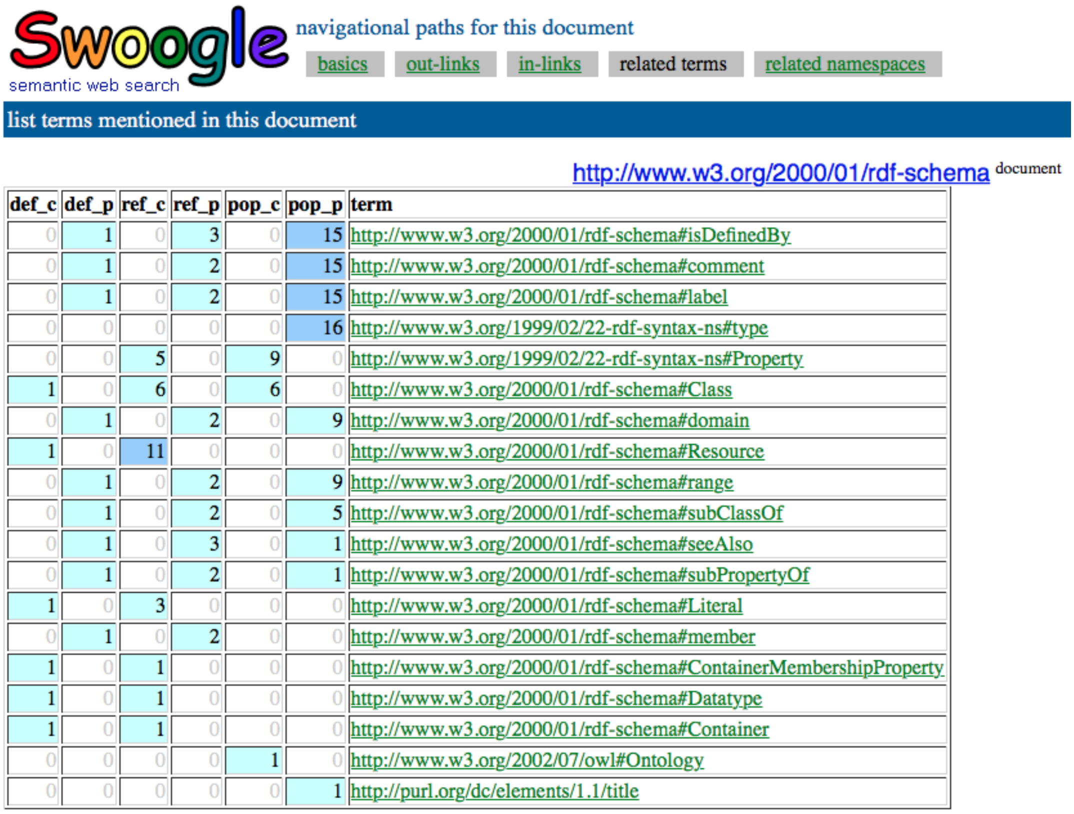
Swoogle displays a list of vocabularies and ontologies that are linked to the searched ontology.
For each one of the vocabularies and ontologies registered on LOV, the user is able to see these relations. Figure 2.8 shows the incoming and outgoing links of the FOAF vocabulary[22]. The difference between Swoogle and LOV is that LOV presents these links graphically and Swoogle in a list, as shown in Figure 2.9.
The OUIP System claims it displays data that are "narrower than" or "broader than" the searched term.
As for the other systems, there is no evidence that they present this feature.

2.2.2 Interfaces

A common practice while developing a system, is to offer an interface for each of the functionalities of the system; but there is the possibility that two features have a common interface.

In this section, an evaluation of the interfaces and how they were implemented in each of the systems is presented. Table 2.4 is a summary of the

[22]Friend of a Friend - `http://xmlns.com/foaf/0.1/`



def_c	def_p	ref_c	ref_p	pop_c	pop_p	term
0	1	0	3	0	15	http://www.w3.org/2000/01/rdf-schema#isDefinedBy
0	1	0	2	0	15	http://www.w3.org/2000/01/rdf-schema#comment
0	1	0	2	0	15	http://www.w3.org/2000/01/rdf-schema#label
0	0	0	0	0	16	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
0	0	5	0	9	0	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
1	0	6	0	6	0	http://www.w3.org/2000/01/rdf-schema#Class
0	1	0	2	0	9	http://www.w3.org/2000/01/rdf-schema#domain
1	0	11	0	0	0	http://www.w3.org/2000/01/rdf-schema#Resource
0	1	0	2	0	9	http://www.w3.org/2000/01/rdf-schema#range
0	1	0	2	0	5	http://www.w3.org/2000/01/rdf-schema#subClassOf
0	1	0	3	0	1	http://www.w3.org/2000/01/rdf-schema#seeAlso
0	1	0	2	0	1	http://www.w3.org/2000/01/rdf-schema#subPropertyOf
1	0	3	0	0	0	http://www.w3.org/2000/01/rdf-schema#Literal
0	1	0	2	0	0	http://www.w3.org/2000/01/rdf-schema#member
1	0	1	0	0	0	http://www.w3.org/2000/01/rdf-schema#ContainerMembershipProperty
1	0	1	0	0	0	http://www.w3.org/2000/01/rdf-schema#Datatype
1	0	1	0	0	0	http://www.w3.org/2000/01/rdf-schema#Container
0	0	0	0	1	0	http://www.w3.org/2002/07/owl#Ontology
0	0	0	0	0	1	http://purl.org/dc/elements/1.1/title

FIGURE 2.9: Example of how Swoogle displays links between vocabularies and ontologies for RDF-Schema vocabulary.

interfaces implemented by each of the reviewed repositories. The interface for SPARQL Endpoint is not present in all of the systems, same as the versioning interface. The most implemented interfaces are Display Metadata and Search Interface, and half of the repositories implement RestFul APIs.

The following subsections introduce a discussion of each of the interfaces mentioned above in detail.

2.2.2.1 SPARQL Endpoint

Offering a SPARQL Endpoint interface to users opens the system to further uses. The user can execute more elaborated and rich queries to find what he/she is looking for.

As displayed in Table 2.4, only two of the metadata repositories, MMI-ORR and LOV, offer this interface to users; Figure 2.10 is an example of how MMI-ORR implements this interface. This interface was implemented using YASGUI[23] or YASGE[24] a client-side JavaScript SPARQL editor.
Additionally, LOV offers a section with query examples, for instance, a query for vocabularies described in a language different to English.

[23] http://about.yasgui.org
[24] http://yasqe.yasgui.org

Repository	SPARQL Endpoint	Display Metadata	Search Interface	Versioning	RestFul API
OUIP	N/A	Yes	Yes	No	Yes
Swoogle	No	Yes	Yes	No	Yes
Oyster	No	N/A	Yes	N/A	N/A
BioPortal	No	Yes	Yes	N/A	Yes
Cupboard	N/A	N/A	N/A	N/A	N/A
MMI-ORR	Yes	Yes	Yes	Yes	N/A
OntoHub	No	Yes	Yes	Yes	N/A
LOV	Yes	Yes	Yes	Yes	Yes

TABLE 2.4: Identification of the interfaces existing in each of the eight repositories.

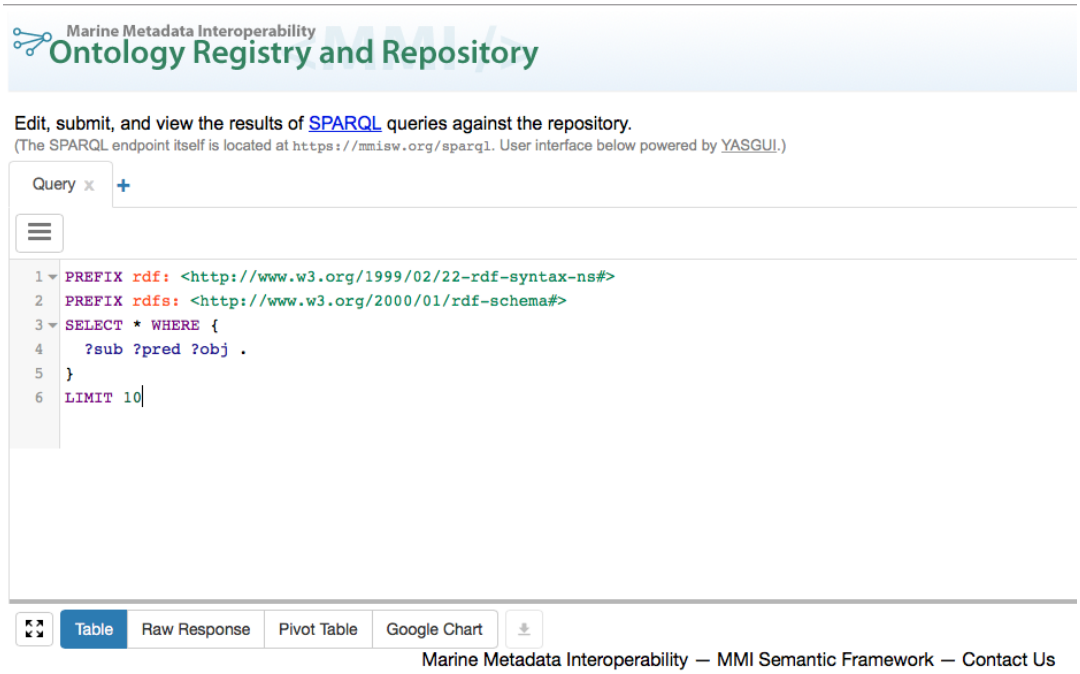


FIGURE 2.10: MMI-ORR SPARQL Endpoint Interface.

Swoogle, Oyster, OntoHub, and BioPortal do not offer a SPARQL Endpoint interface.

2.2.2.2 Web Search

Having an interface for searching terms, vocabularies and ontologies allow all users to use the system, not only those who know SPARQL rules can use it.

This interface could be developed simple, like LOV, Swoogle or MMI-ORR, in which they offer a field to type word(s) and search for data that contain these words. Alternatively, larger like BioPortal's interface (Figure 2.11), that offers "advanced search" with further options to exploit, nonetheless BioPortal also offers a simple search interface.
Even though the OUIP System does not offer SPARQL Endpoint interface, it does offer an interface for searching keywords.
As mentioned before, Swoogle offers an interface for searching terms, ontologies and documents.

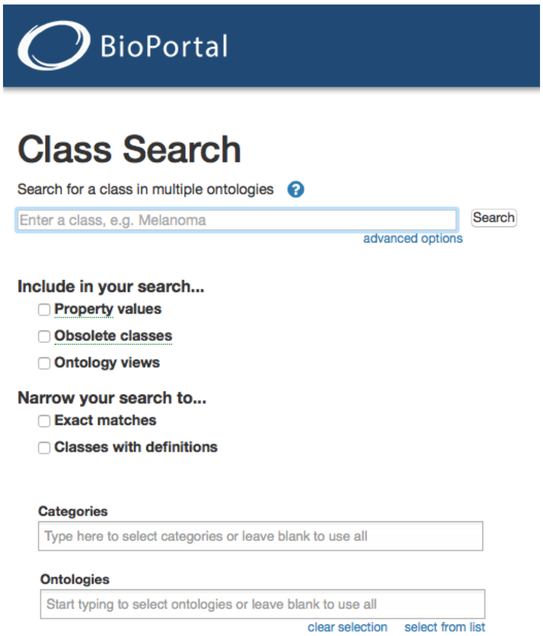


FIGURE 2.11: BioPortal's advanced search options.

Moreover, Oyster offers an interface for searching the repository by typing keywords in the search fields, language, ontology name, status, among others.

2.2.2.3 Display Metadata

An interface in which only metadata about a particular vocabulary is displayed is important. In it, users can see metadata aspects like title, creators, year of creation, and so on, of a specific vocabulary. This interface can be designed in different ways, as a table, or a list; but the important part is that the data are displayed clearly.
In Table 2.4 is possible to see that most of the systems have developed this interface.

The metadata displayed for each vocabulary depends on the data model of the system, for example, Figure 2.12 shows the stored metadata in MMI-ORR, the metadata is divided into the following categories, General, Usage/License/Permissions, and more.
Swoogle presents the metadata in a simple way, and similar to MMI-ORR, it divides the metadata into categories, basics, out-links, in-links, related terms, among others. If a user wants the metadata presented in RDF, he/she can select the "RDF version" link. The metadata will be displayed in RDF/XML format with its respective vocabularies and properties.
Oyster displays the metadata of the selected ontology, in a separate component. The metadata is shown in RDF/XML format.
LOV has developed this interface to make it more appealing to users. It is divided into components that represent the Metadata. The incoming or outgoing links, the statistics for representing the number of classes, properties,

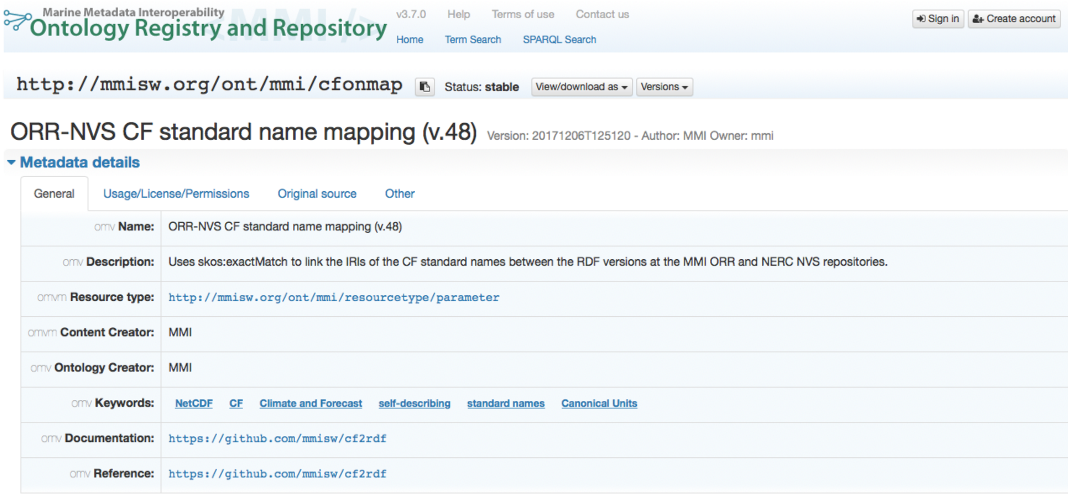


FIGURE 2.12: MMI-ORR metadata display interface.

FIGURE 2.13: Version keeping interface in LOV for vocabulary SKOS.

datatypes and instances a vocabulary has, are some of the components. OntoHub offers an interface for displaying metadata but seems like their primary goal is to display information about the ontologies' data (classes, properties, etc.).

2.2.2.4 Versioning

As a user, being able to see versions of a particular vocabulary or ontology, helps understand if from one version to another there have been changes in terms, or whether there has been added a term more fitting to a specific subject.

LOV allows users to see in a timeline (Figure 2.13), the versions of each vocabulary and ontology, these versions can be downloaded by clicking on them. On the other hand, MMI-ORR displays near the name the version of the vocabulary or ontology displayed plus the status of each vocabulary/ontology, letting users know whether it is stable or not.
Version control of each ontology is possible in OntoHub, locally with a Git repository or online via the browser. As shown in Table 2.4 the other systems do not consider versioning.

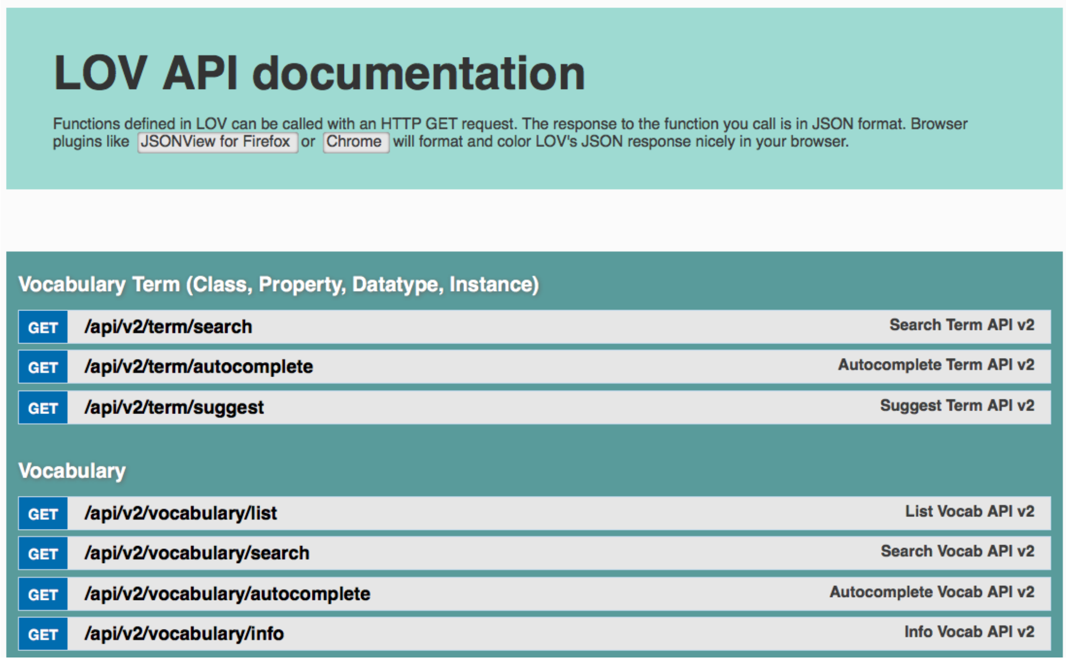


FIGURE 2.14: List of APIs that LOV offers.

2.2.2.5 RESTful API

RESTful APIs or web services interconnect systems, requesting access and manipulation of information.

LOV offers functions over HTTP GET requests, which means that other systems can only request information and cannot manipulate or alter information stored in LOV. The information that can be requested is about Vocabularies, Terms of specific vocabulary and Agents. In LOV, agents refer to organizations or people who have created or contributed to vocabularies and ontologies.
Figure 2.14 presents the APIs that LOV offers. By clicking on each of the boxes, LOV presents which parameters and possible values are necessary for calling each API. For example, Figure 2.15 is an example of the parameters necessary for calling the search terms API. This API requires the parameters `q` with the search criteria, `type` with the type of term to be searched, which possible values are, class, property, datatype, and instance; `vocab` specifying the one vocabulary to search on.

As documented in `http://data.bioontology.org/documentation`, BioPortal offers HTTP GET and POST operations, for searching terms, for example, when searching for the term "melanoma", the user can type the following URL `http://data.bioontology.org/search?q=melanoma`.
Similar to LOV, this API has a list of parameters for refining the search; using the keywords, `ontologies=`, `include=`, among others, the user can specify in which ontologies to search, and what metadata to include.
BioPortal recommends ontologies to the users based on a list of keywords provided as input. For example, `http://data.bioontology.org/`

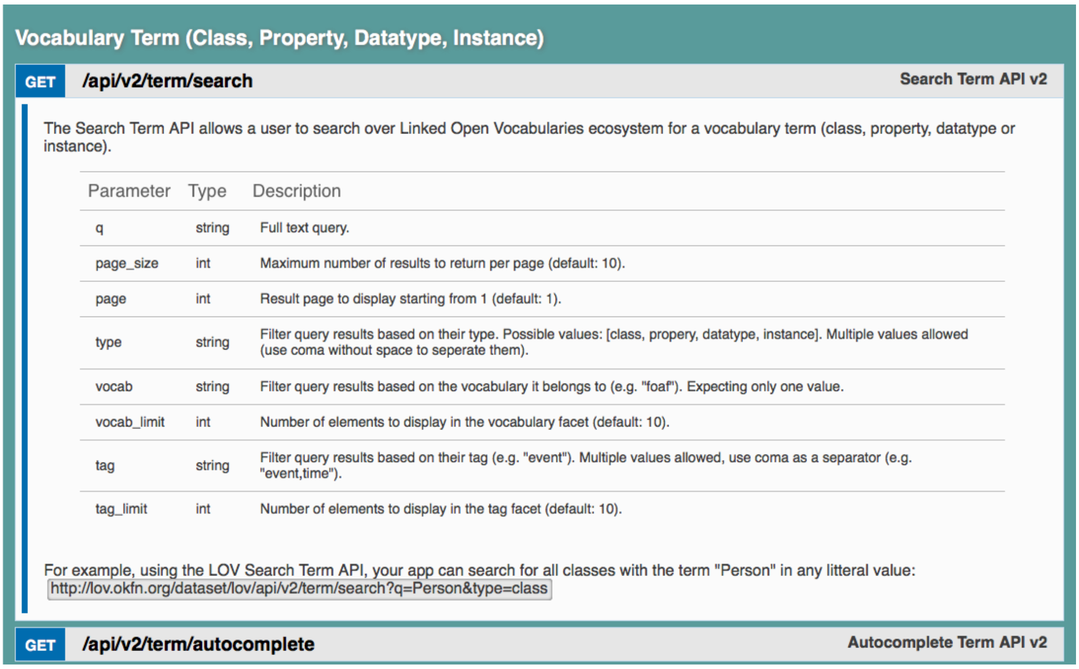


FIGURE 2.15: Example of an API from LOV, displaying necessary parameters and possible values.

recommender?input=Melanoma%20is%20a%20malignant%20tumor%20of%20melanocytes%20which%20are%20found%20predominantly%20in%20skin%20but%20also%20in%20the%20bowel%20and%20the%20eye will recommend the user ontologies that are appropriate for the text "Melanoma is a malignant tumor of melanocytes which are found predominantly in skin but also in the bowel and the eye". Figure 2.16, is a snippet of the result the system will display, it is a JSON file with the recommended ontologies.

Swoogle also allows Web Services, the documentation on how to use them can be found in http://swoogle.umbc.edu/2006//index.php?option=com_swoogle_manual&manual=search_overview.

2.2.3 Data Model

A metadata repository is not only composed of the features it offers to the users and how they are implemented, but it also has an underlying data model. The data model is the structure of how the data are being stored, which properties are being used for describing the metadata of a vocabulary or ontology.

Metadata	Properties used by the repositories	
Namespace	Swoogle	owl:usesNamespace
	Oyster	omv:ontologyURL
	MMI-ORR	vann:preferredNamespacePrefix

Continue on next page

Table 2.5 – continued from previous page

Metadata	Properties used by the repositories	
	LOV	vann:preferredNamespacePrefix
		vann:preferredNamespaceUri
Title	Oyster	omv:implementationName
	MMI-ORR	dct:title
		omv:name
	LOV	dct:title
Description	Swoogle	rdfs:comment
	Oyster	omv:implementationDescription
	MMI-ORR	dct:description
	LOV	dct:description
Creator	Oyster	omv:implementationCreator
	MMI-ORR	omv:hasCreator
		omvm:hasContentCreator
	LOV	dct:creator
		dct:contributor
		dct:publisher
Language	LOV	dct:language
Creation	Swoogle	swoogle:hasDateDiscovered
	Oyster	omv:creationDate
	MMI-ORR	dct:created
	LOV	dct:issued
Modified	Swoogle	swoogle:hasDateLastmodified
	LOV	dct:modified
Keywords	MMI-ORR	omv:keywords
	LOV	dcat:keyword
Version	Swoogle	owl:versionInfo
		daml:versionInfo
	Oyster	omv:versionInfo
	MMI-ORR	owl:versionInfo
	LOV	dcat:distribution
Homepage	MMI-ORR	omv:documentation
	LOV	foaf:homepage
Comments	MMI-ORR	rdfs:comment
	LOV	rev:hasReview
Rights	MMI-ORR	dct:rights
License	MMI-ORR	dct:license
	LOV	dct:license

TABLE 2.5: Vocabularies and properties used by each repository to describe metadata.

Vocabulary prefixes: dct = `http://purl.org/dc/terms/`, owl = `http://www.w3.org/2002/07/owl#`, rdfs = `http://www.w3.org/2000/01/rdf-schema#`, vann = `http://purl.org/vocab/vann/`, dcat = `http://www.w3.org/ns/dcat#distribution`, omv = `http://omv.ontoware.org/2005/05/ontology#`, swoogle = `http://daml.umbc.edu/ontologies/webofbelief/1.4/swoogle.owl#`, daml = `http://www.daml.org/2001/03/daml+oil#`

```
[
  - {
      evaluationScore: 0.632,
    - ontologies: [
        - {
            acronym: "NCIT",
            @id: http://data.bioontology.org/ontologies/NCIT,
            @type: http://data.bioontology.org/metadata/Ontology,
          - links: {
                submissions: http://data.bioontology.org/ontologies/NCIT/submissions,
                properties: http://data.bioontology.org/ontologies/NCIT/properties,
                classes: http://data.bioontology.org/ontologies/NCIT/classes,
                single_class: http://data.bioontology.org/ontologies/NCIT/classes/{class_id},
                roots: http://data.bioontology.org/ontologies/NCIT/classes/roots,
                instances: http://data.bioontology.org/ontologies/NCIT/instances,
                metrics: http://data.bioontology.org/ontologies/NCIT/metrics,
                reviews: http://data.bioontology.org/ontologies/NCIT/reviews,
                notes: http://data.bioontology.org/ontologies/NCIT/notes,
                groups: http://data.bioontology.org/ontologies/NCIT/groups,
                categories: http://data.bioontology.org/ontologies/NCIT/categories,
                latest_submission: http://data.bioontology.org/ontologies/NCIT/latest_submission,
                projects: http://data.bioontology.org/ontologies/NCIT/projects,
                download: http://data.bioontology.org/ontologies/NCIT/download,
                views: http://data.bioontology.org/ontologies/NCIT/views,
                analytics: http://data.bioontology.org/ontologies/NCIT/analytics,
                ui: http://bioportal.bioontology.org/ontologies/NCIT,
              - @context: {
                    submissions: http://data.bioontology.org/metadata/OntologySubmission,
```


FIGURE 2.16: JSON file of the result from executing the recommender API of BioPortal.

There exist specialized vocabularies and ontologies for describing metadata, for example, Dublin Core Metadata Initiative[25] [27] aims to offer a metadata format that can be used by any vocabulary in any domain. Equal to this, there exist domain-specific vocabularies, e.g., Semantic Sensor Network[26] is an ontology for describing sensors and their measurements in the marine domain.

Metadata repositories do not restrict themselves to just one vocabulary or ontology for describing their metadata; they use a combination of them. Determining which terms from which vocabularies to use is not an easy task, due to the possibility of having more than one term, for the same purpose. Different vocabularies that aim to describe the same terminology could present distinct naming. For example, for describing the title of a vocabulary or ontology, users can use, either `dcterms:title` or `rdfs:label`.

In this section, a comparison of the repositories' data models is presented. Table 2.5 details which metadata is stored by the surveyed repositories with their respective vocabularies and properties.
The properties most covered for each of the vocabularies/ontologies stored in the metadata repositories are the namespace, title, description, creator, creation date and version. Which led us to think that these properties are mandatory to be collected about all vocabularies present in a metadata repository.

[25] http://dublincore.org/documents/dcmi-terms/
[26] https://www.w3.org/TR/vocab-ssn/

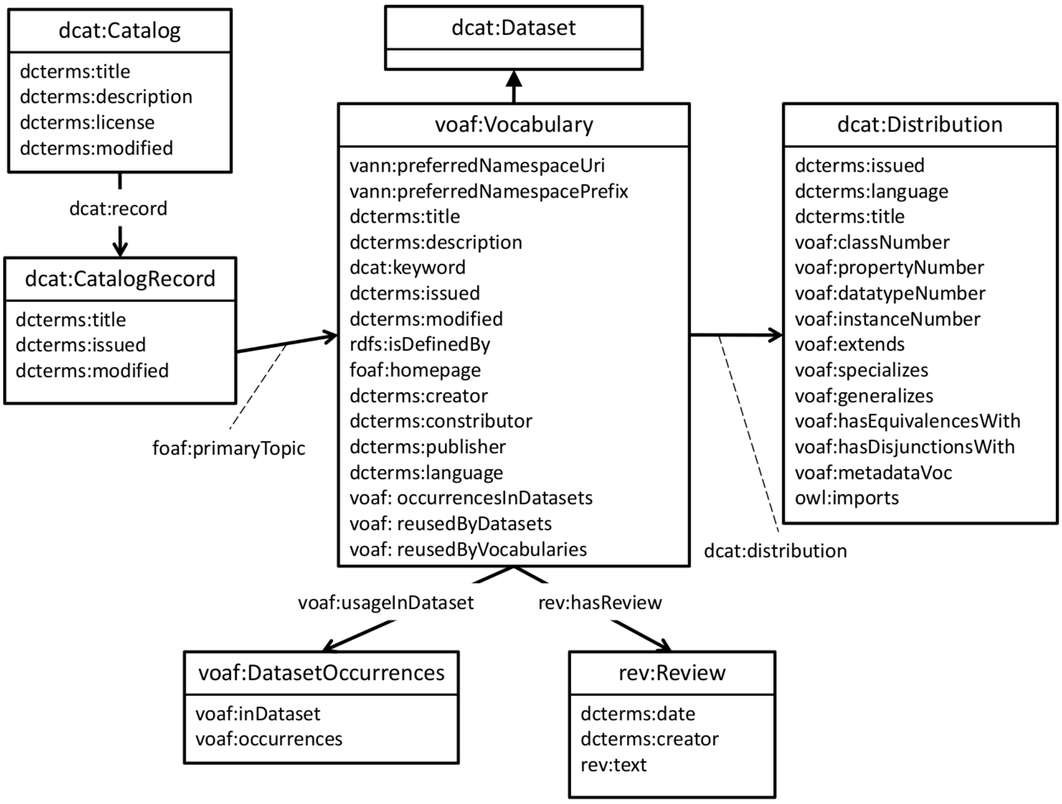


FIGURE 2.17: UML class diagram representing LOV data model. Image from Vandenbussche, Pierre-Yves and Atemezing, Ghislain A and Poveda-Villalón, María and Vatant, Bernard. Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web. Semantic Web, 8(3): 437–452, 2017.

Not all repositories present their data model openly; for example, in BioPortal, by looking at the Web page one can infer which metadata is stored, but semantically it is not possible to see which Semantic vocabularies and its properties are being used. On the other hand, Swoogle mentions in the publication just three properties used to store the metadata. However, in order to be able to see which vocabularies and properties the repository uses, it was necessary to read the metadata presented in RDF/XML link on the Web Page.

LOV is one of the repositories that describe its data model [5]. Figure 2.17 shows which metadata LOV stores and which vocabularies and properties are being used. LOV uses a combination of approximately six (6) vocabularies for extensive metadata description. Furthermore, LOV stores metadata about each vocabularies data (Figure 2.6); the number of classes, number of properties, and relationships between other vocabularies and ontologies, e.g., extensions and specializations.

Oyster stores the vocabularies and ontologies metadata using the Ontology Metadata Vocabulary (OMV). Similar to LOV, it also stores metadata

Metadata	Oyster	LOV
Class Number	omv:numClasses	voaf:classNumber
Property Number	omv:numProperties	voaf:propertyNumber
Datatype Number	–	voaf:datatypeNumber
Instance Number	–	voaf:instanceNumber
Extends	–	voaf:extends
Specializes	–	voaf:specializes
Equivalences	–	voaf:hasEquivalencesWith
Disjunctions	–	voaf:hasDisjunctionsWith

omv = http://omv.ontoware.org/2005/05/ontology#, voaf = http://purl.org/vocommons/voaf#

TABLE 2.6: Vocabularies and properties used by each repository to describe metadata.

about the vocabularies data, number of classes and number of properties.

For the other repositories, it is difficult to evaluate their data model, due to lack of information, either on their respective paper or their online repository.

Chapter 3

Methodology for Creating a Metadata Repository

As seen in the previous chapter, there exists a large number of vocabularies in different domains, but there are not enough domain-specific metadata repositories for organizing and managing them. Different communities want to have vocabularies and ontologies related to their domain in one common place to ease the search, use, and share of these vocabularies and ontologies for their use cases.

This chapter presents BigDataOcean Metadata Repository's specifications and implementation. Then from the experience gained, a methodology proposal is offered, where it intends to give a step-by-step list of what to do and how to decide to create or extend a metadata repository, a list of vocabularies and ontologies and a list of metadata properties that can be used to describe any vocabulary or ontology.

3.1 Case Study: BigDataOcean Metadata Repository

BigDataOcean[1] is a European Project that "*aims to enable maritime big data scenarios for EU-based companies, organizations and scientists, through a multi-segment platform that will combine data of different velocity, variety, and volume under an inter-linked, trusted, multilingual engine*"[28].

BigDataOcean Metadata Repository (BDO) forms part of the infrastructure to access and share Linked Big Data vocabularies and ontologies metadata.

BigDataOcean has four pilot cases [28]:

1. Fault Prediction and Proactive Maintenance need to describe data about unpredicted damages and mechanical failures of vessels, and environmental damages caused,

[1] http://www.bigdataocean.eu/site/

	Big Data Ocean Metadata Repository Requirements
FR	Insert new vocabularies/ontologies in the Metadata Repository
FR	Import existing vocabularies/ontologies into the Metadata Repository
FR	Delete vocabularies/ontologies from the Metadata Repository
FR	Insert metadata about vocabularies/ontologies in the Metadata Repository
FR	Describe vocabularies/ontologies in the Metadata Repository using ontologies
FR	Search vocabularies/ontologies in the Metadata Repository based on different criteria and keyword
FR	Evaluate SPARQL queries over the Metadata Repository to retrieve classes and properties of vocabularies/ontologies
NFR	Ensure persistence of the Metadata Repository
NFR	Ensure web-based access and availability of the Metadata Repository
FR	Compute statistics about the Repository vocabularies / ontologies
FR	Search pilots using particular vocabularies / ontologies
FR	Link Repository vocabularies/ontologies with similar existing vocabularies/ontologies
FR	Keep track of changes and versions of Repository vocabularies/ontologies
FR	Provide a Question Answering system on top of the Metadata Repository

TABLE 3.1: List of collected functional and nonfunctional requirements of Big Data Ocean Metadata Repository

2. Mare Protection requires the description of atmospheric, wave and hydrodynamical data, combined with location, rate, and characteristics of an oil spill,

3. Maritime Security and Anomaly Detection, need data about security, events, and threats in the maritime environment to identify patterns that will impact security, economy, and environment, such as terrorism, illegal trafficking, and fishing, and

4. Wave Power aims to contribute to wave energy industry by offering the ability to predict the best locations, the expected energy production, equipment costs, and environmental impact. For this reason, storing environmental and geophysical data coming from vessels, buoys are important.

3.1.1 Requirements

BigDataOcean Metadata Repository has a list of functional and non-functional requirements [29]. In order to collect the requirements, during the elicitation period, it was necessary to collect two types of information from the pilots' partners. This information concerns the data sources, own and external data to be used by the stakeholders associated with each pilot, and through the form "Data Source Definition Form" annexed in [30] identify possible Linked Data vocabularies that can be used to map the data sources. The next step was the phase of prioritizing the requirements, followed by the study of existing the metadata repositories comparing their functionalities to the list of requirements of Big Data Ocean Metadata Repository. Table 3.1

is a list of some of the functional and nonfunctional requirements collected from the steps mentioned before.
This study resulted in Linked Open Vocabularies (LOV) to be the metadata repository most fitted to extend by cause of most of the Big Data Ocean repository requirements were already developed in LOV. Additionally, LOV is open-source, and its code is published on GitHub.

In this section, the requirements CRUD operations, search vocabularies, and ontologies, SPARQL queries, compute statistics and visualization of connectivity between vocabularies and ontologies are discussed. The Table 3.2 displays the reused and adapted functionalities from LOV repository and the added requirements in BigDataOcean Metadata Repository.

BDO Metadata Repository Requirements	Features from LOV		
	Reused	Adapted	Added
CRUD vocabularies and ontologies from the metadata repository		✓	
Search vocabularies and terms in the metadata repository	✓		
SPARQL queries over the metadata repository		✓	
Compute statistics about the repository vocabularies and ontologies		✓	
Visualize connectivity among vocabularies and ontologies	✓		
CRUD and search pilots using particular vocabularies/ontologies			✓

TABLE 3.2: List of the requirements reused and adapted from LOV, and added in BigDataOcean Metadata Repository.

CRUD vocabularies and ontologies from the metadata repository

For Big Data Ocean project it is essential to be able to insert and import new vocabularies/ontologies, modify and delete metadata of existing vocabularies/ontologies. Figure 3.1 shows the interface for inserting, importing or editing/modifying the metadata of a particular vocabulary, in this case for The Geonames ontology (gn).
LOV already implemented these functionalities, but BDO data model was extended and stores more information related to each vocabulary and ontology. It was necessary to extend the interfaces for allowing the addition of more metadata, like pilots in which one vocabulary could be used.

Search vocabularies and terms in the metadata repository

LOV offers the ability to search for vocabularies/ontologies and terms given a keyword. This feature was already developed, and it was not required to extend it. Figure 3.2 shows the interfaces for searching vocabularies and ontologies and terms in BDO Metadata Repository.

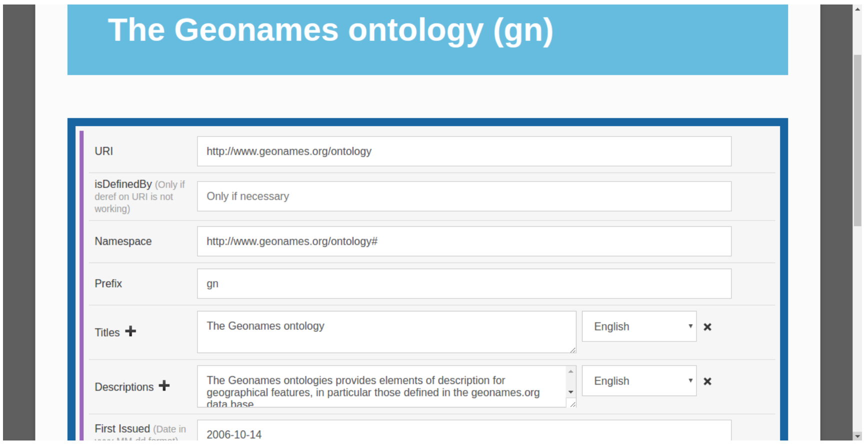


FIGURE 3.1: Interface for inserting, importing or modifying a vocabulary metadata in BDO Metadata Repository.

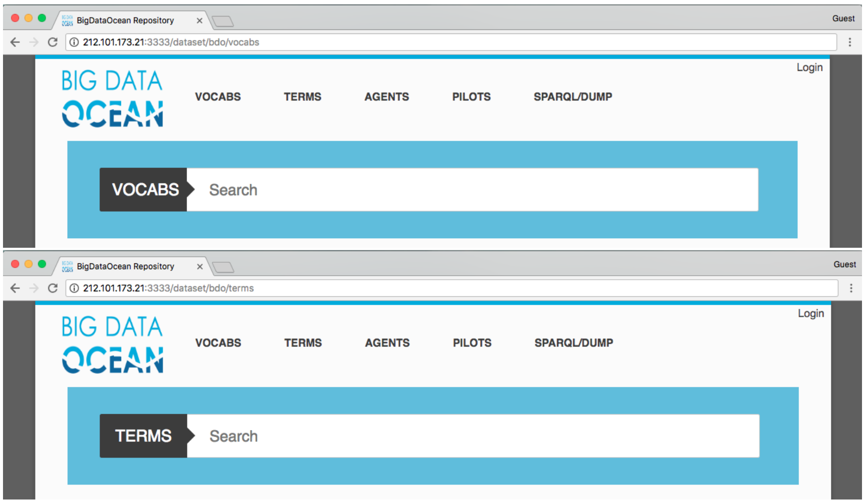


FIGURE 3.2: Searching vocabularies and ontologies interfaces in BDO Metadata Repository.

The search interface and feature are simple; the user can type any keyword and search, if it is necessary to refine the search further, the user can select from the right hand of the page, a specific Tag, Pilot or Language. Figure 3.3 is an example of searching vocabularies that contains the keyword "sensor".

SPARQL queries over the metadata repository

As mentioned in Chapter 2 Section 2.2.1, allowing users with SPARQL query rules knowledge perform queries, opens the system to advanced searches. LOV developed this feature and was adopted by BDO Metadata Repository, only adjusting the entry point to the BDO Metadata Repositories' triplestore. Furthermore, because of LOV's Question and Answering feature, BDO also offers the functionality to perform SPARQL queries over Q&A. Figure 3.4 is the interface of BDO Metadata Repository's SPARQL endpoint.

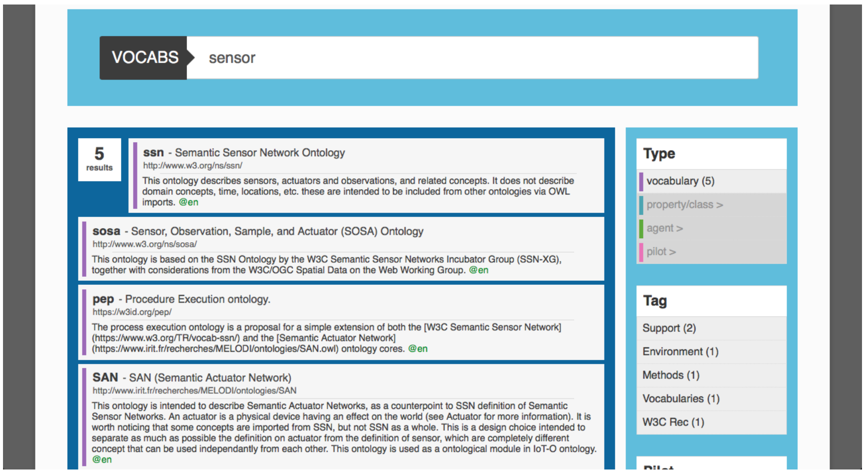


FIGURE 3.3: Example of searching vocabularies with the keyword "sensor" in BDO Metadata Repository.

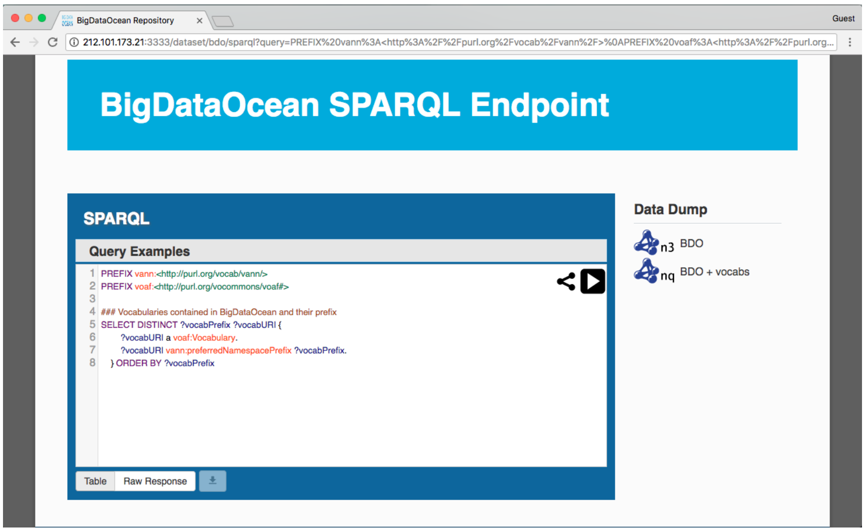


FIGURE 3.4: SPARQL endpoint for querying vocabularies and their metadata in BDO Metadata Repository.

Compute statistics about the repository vocabularies and ontologies

This functionality was entirely adopted from LOV. Functionally in BDO metadata repository was not changed, but the interface was adapted so when the users want to see the terminology of the displayed vocabulary/ontology, he/she can click on the statistics chart, and they will be redirected to the terms page showing only the terms of the vocabulary.
In Figure 3.5 it is portrait the metadata and statistics of the Semantic Sensor Network Ontology (SSN).

Visualize connectivity among vocabularies and ontologies

By showing the connections among vocabularies and ontologies stored in the metadata repository, the user can see if there exists another vocabulary that will help him/her be more specific or general while describing their dataset

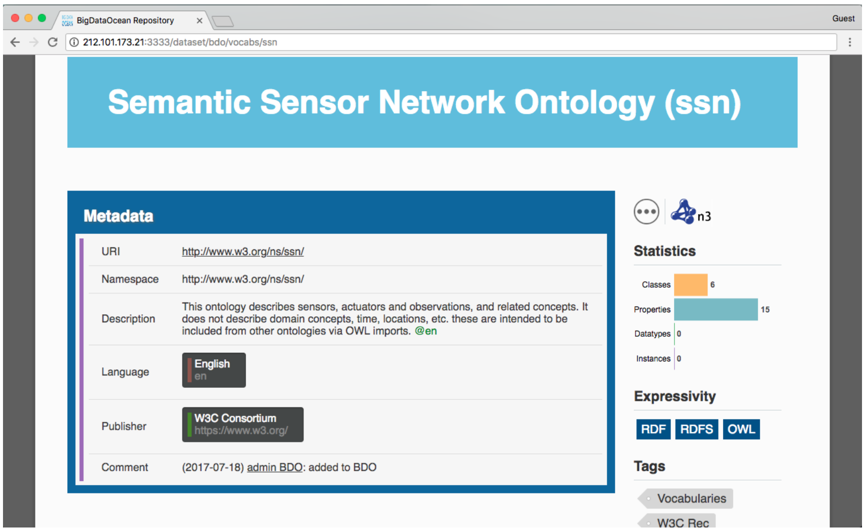


FIGURE 3.5: Metadata and statistics of the Semantic Sensor Network Ontology (SSN) displayed in BDO Metadata Repository.

or vocabulary/ontology. Figure 3.6 shows the connections of SKOS Vocabulary in BDO Metadata Repository.

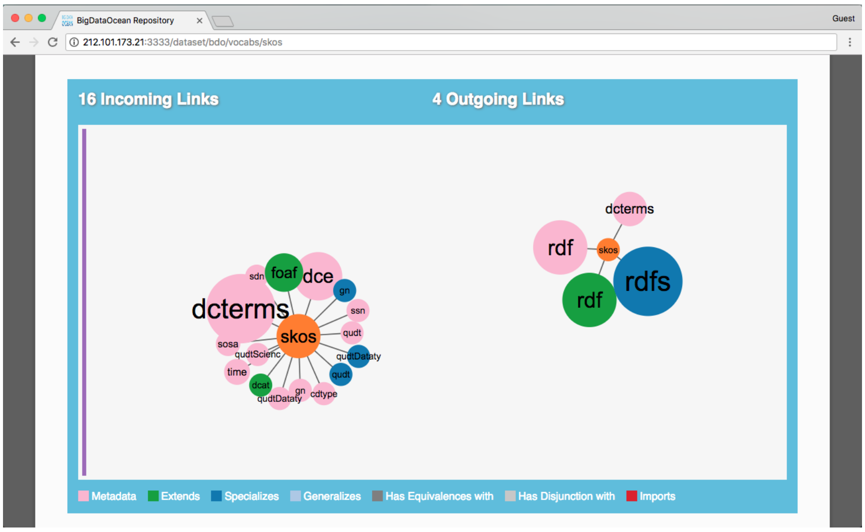


FIGURE 3.6: BDO Metadata Repository visualization of vocabularies and their connections.

CRUD and search pilots using particular vocabularies/ontologies

Additionally, as mentioned earlier, BDO manage four different pilots. Each of these pilots needs different vocabularies for describing their datasets accurately.
These functionalities, CRUD, and search for pilots were added for BDO Metadata Repository. LOV does not manage any pilot. Thus it was necessary to develop it.

Figure 3.7 shows which vocabularies could be used for describing datasets of the Pilot I - Fault Prediction and Proactive Maintenance.

FIGURE 3.7: Vocabularies related to Pilot I - Fault Prediction and Proactive Maintenance in BDO Metadata Repository.

3.1.2 Implementation

BigDataOcean Metadata Repository was adapted from LOV, in the Frontend the colors where changed, and the tab "PILOTS" was added. In the Backend, it was necessary to implement functions for add, modify, and delete pilots; and to extend the system's API, by implementing the pilots APIs that will allow users to query the metadata repository.

3.1.2.1 Technical Background

Its Frontend was developed using NodeJs[2], Jade[3], JavaScript[4], HTML and CSS.
The Backend was developed in Java and Maven[5].
For data management, it uses MongoDB[6] for storing metadata, ElasticSearch[7] as search engine and Apache Jena Fuseki[8] for SPARQL server and TripleStore.

3.1.2.2 Data Model

The BigDataOcean Metadata Repository data model stores almost the same data as Linked Open Vocabularies (LOV) does. The most significant

[2] https://nodejs.org/en/
[3] https://www.jadeworld.com/developer-center
[4] https://www.javascript.com
[5] https://maven.apache.org
[6] https://www.mongodb.com
[7] https://www.elastic.co
[8] https://jena.apache.org/documentation/fuseki2/

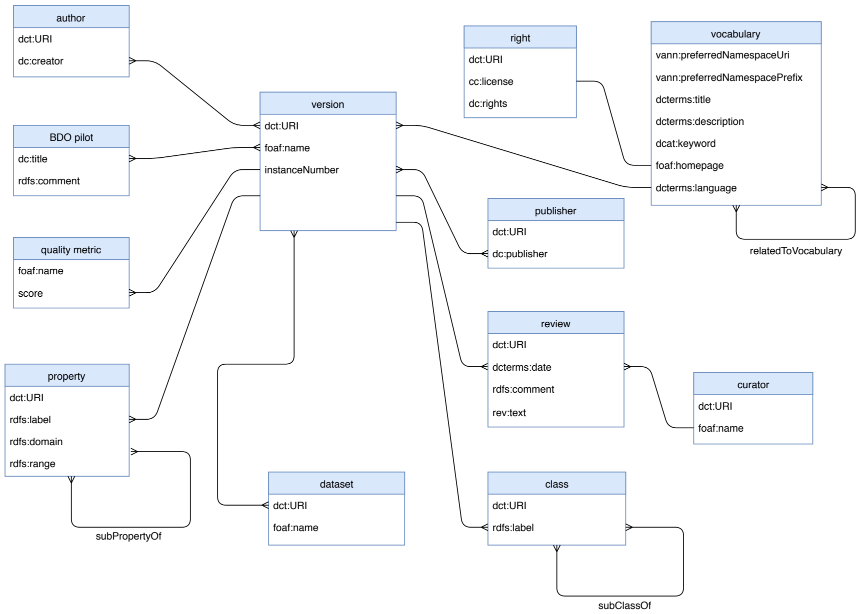


FIGURE 3.8: BigDataOcean Metadata Repository entity-relationship diagram, representing the entities, properties and their relations captured in the system. Image from BigDataOcean Deliverables. D3.1 BigDataOcean Linked Data Vocabularies and Metadata Repository Architecture, http://www.bigdataocean.eu/site/wp-content/uploads/2016/12/BigDataOcean_Linked_Data_Vocabularies_and_Metadata_Repository_Architecture_v1.00.pdf.

difference between these data models is that BDO Metadata Repository additionally stores information about the project pilots.
Figure 3.8 is the Entity-Relationship diagram of BigDataOcean Metadata Repository, representing the entities, properties and their relations captured in the system.

When describing vocabularies and ontologies in the BigDataOcean Metadata Repository, the following entities are used.

- vocabulary: information about a vocabulary including title, description, keywords, language, etc.;
- version: version of a vocabulary – a vocabulary may have one or more versions;
- author: creator(s) of a vocabulary;
- publisher: publisher(s) of a vocabulary;
- curator: curator(s) of a vocabulary;
- review: a curator provides reviews for a specific version of the vocabulary;
- right: a vocabulary is associated with a license;

- property: metadata about vocabulary properties;
- class: metadata about vocabulary classes;
- quality metric: metadata describing the quality of the vocabulary according to metrics;
- dataset: dataset related to a vocabulary - it can be related to one or more vocabularies;
- BDO pilot: BigDataOcean pilot - the pilot can be related to one or more vocabularies.

BigDataOcean Metadata Repository is available online on http://212.101.173.21:3333/dataset/bdo/. Figure 3.9 shows the number of vocabularies/ontologies currently indexed in BigDataOcean metadata repository.

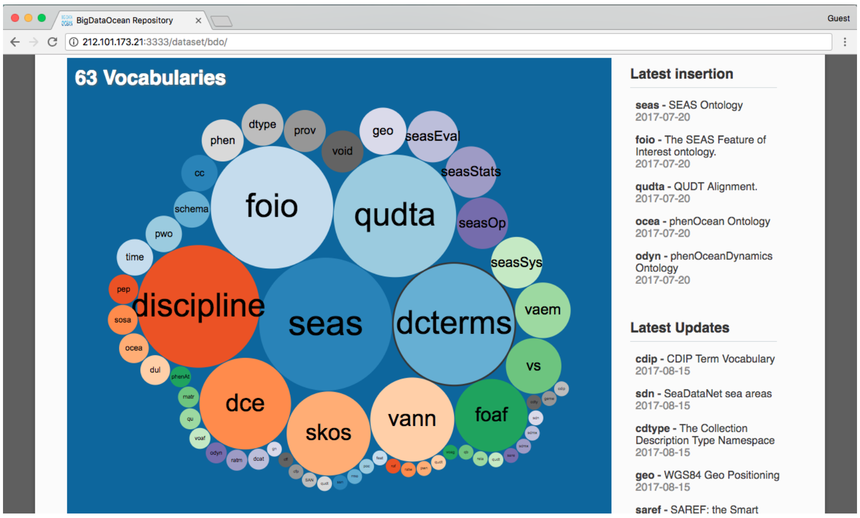


FIGURE 3.9: BigDataOcean Metadata Repository - vocabularies

3.2 Methodology

In this section, a methodology proposal for extending or creating a metadata repository is described. This methodology was built from the experience gained while extending BigDataOcean Metadata Repository from Linked Open Vocabularies (LOV), presented in the previous section.

Developing from scratch a metadata repository requires a good understanding of the functionalities the system will offer and a detailed data model. This development will demand time and extensive testing; whereas extending an already existing metadata repository will require time for adapting existing and developing new features, plus time for testing. How to decide what is better for the project?

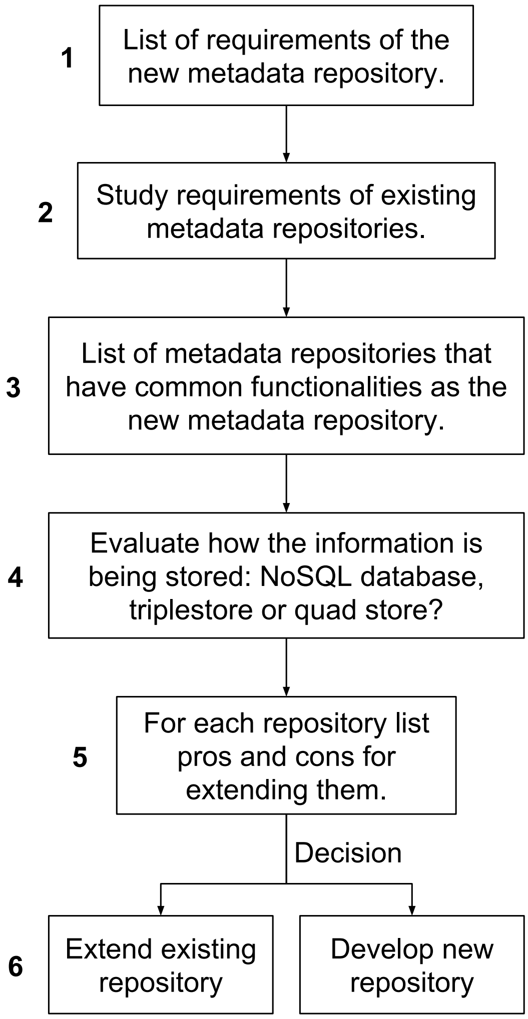


FIGURE 3.10: Step-by-step workflow of the methodology for deciding when to extend an existing metadata repository or to develop a new metadata repository.

The following is a proposal on how to decide whether to create or extend a metadata repository, and Figure 3.10 depicts the step-by-step workflow of the proposal.

- First, the software developer or project manager needs to have a clear view of the functionalities to be implemented in the system, will it allow to CRUD the system? Will users be able to search the system using keywords?
- Then, evaluate if there already exists a metadata repository that has the majority of these functionalities and whether it is easy to extend or not, in order to satisfy the list of functionalities the new system requires.
- After finishing the evaluation of existing metadata repositories, it is possible that there exists more than one repository fitting the needs

of the project. To further decide on extending or developing a new metadata repository, it is recommendable to evaluate how the information is stored. Knowledge graphs can be stored on Triplestores, Quad stores or NoSQL databases. However, some triplestores and quad stores offer reasoners and/or SPARQL endpoints, among other features. `https://www.w3.org/RDF/` offers a list of Triplestores tools.

- Having more information will facilitate the decision making; "extending this or that repository will ease the development of the tool because, ..." (and list of pros and cons).

Functional Requirements		
Requirements	Must have	Nice to have
* Insert, delete and update vocabularies and ontologies	✓	
* Import published vocabularies and ontologies		✓
Compute statistics about vocabularies and ontologies		✓
Store and visualize links between vocabularies and ontologies		✓
Keep track of changes (versioning)	✓	
Evaluate SPARQL queries over the Metadata Repository		✓
Search vocabularies and ontologies based on keywords	✓	
Offer SPARQL Endpoint for query evaluation		✓
Offer a Question and Answering system		✓

* If the system have one requirement, it would be nice to have the other and viceversa.

TABLE 3.3: Functional Requirements recommended that a metadata repository might offer.

Non-Functional Requirements		
Requirements	Must have	Nice to have
Ensure persistence	✓	
Ensure Web-based access	✓	
Allow extensibility		✓

TABLE 3.4: Non-Functional Requirements recommended that a metadata repository might offer.

Tables 3.3 and 3.4 are a recommendation of functional and non-functional requirements a metadata repository might offer, based on the literature survey in Chapter 2.
While studying the metadata repositories, all of them described functionalities and how the system should behave; but a few or none of them mentioned the non-functional requirements of their systems.
The proposed non-functional requirements were taken from [31], where it offers the best practices for software engineering.
The functional requirements are the union of the functionalities developed by the metadata repositories studied in the literature survey, giving the opportunity to have a complete system. In which users with different backgrounds, are able to use the system without extensive technical course or support.

If the system has the functionality insert, delete and update vocabularies and ontologies, it would be nice that the system has the functionality to import published vocabularies and ontologies and the other way around.

Table 3.5 is a list of several vocabularies and ontologies specialized on defining properties and classes for describing other Semantic Web documents, vocabularies, and ontologies.

Vocabulary's Name	Link
Dublin Core Metadata Initiative, Metadata terms	http://purl.org/dc/terms/
Web Ontology Language (OWL)	http://www.w3.org/2002/07/owl#
RDF Schema (RDFS)	http://www.w3.org/2000/01/rdf-schema#
Ontology Metadata Vocabulary (OMV)	http://omv.ontoware.org/2005/05/ontology#
DARPA Agent Markup Language (DAML)	http://www.daml.org/2001/03/daml+oil#
RDF Review Vocabulary	http://purl.org/stuff/rev#
A vocabulary for annotating vocabulary descriptions (VANN)	http://purl.org/vocab/vann/
Data Catalog Vocabulary (DCAT)	http://www.w3.org/ns/dcat#distribution
swoogle	http://daml.umbc.edu/ontologies/webofbelief/1.4/swoogle.owl#

TABLE 3.5: List of vocabularies recommended for describing and annotating metadata.

For a metadata repository data model, it is recommended to store as much metadata as possible about each vocabulary and ontology. Table 3.6 shows a list of metadata that should be stored with a list of alternative properties; for example for specifying who created certain vocabulary or ontology, it is possible to use one of these properties `omv:implementationCreator`, `omv:hasCreator`, `omvm:hasContentCreator` or `dct:creator`.

Metadata	Alternatives Properties
Prefix	vann:preferredNamespacePrefix
Namespace	owl:usesNamespace
	omv:ontologyURL
	vann:preferredNamespacePrefix
	vann:preferredNamespaceUri
Title	omv:implementationName
	dct:title
	omv:name
Description	rdfs:comment
	omv:implementationDescription
	dct:description
Creator	omv:implementationCreator
	omv:hasCreator
	omvm:hasContentCreator
	dct:creator
Contributor	dct:contributor
Publisher	dct:publisher

Continue on next page

Table 3.6 – continued from previous page

Metadata	Alternatives Properties
Language	dct:language
Creation date	swoogle:hasDateDiscovered
	omv:creationDate
	dct:created
	dct:issued
Modification date	swoogle:hasDateLastmodified
	dct:modified
Keywords	omv:keywords
	dcat:keyword
Version	owl:versionInfo
	daml:versionInfo
	omv:versionInfo
	dcat:distribution
Homepage	omv:documentation
	foaf:homepage
Comments	rdfs:comment
	rev:hasReview
Rights	dct:rights
License	dct:license
Number of classes	omv:numClasses
	voaf:classNumber
Number of properties	omv:numProperties
	voaf:propertyNumber
Number of datatype	voaf:datatypeNumber
Number of instances	voaf:instanceNumber
Extends	voaf:extends
Specializes	voaf:specializes
Equivalence	voaf:hasEquivalencesWith
Disjunctions	voaf:hasDisjunctionsWith

TABLE 3.6: List of alternative properties for describing metadata.

Vocabulary prefixes: dct = `http://purl.org/dc/terms/`, owl = `http://www.w3.org/2002/07/owl#`, rdfs = `http://www.w3.org/2000/01/rdf-schema#`, vann = `http://purl.org/vocab/vann/`, dcat = `http://www.w3.org/ns/dcat#distribution`, omv = `http://omv.ontoware.org/2005/05/ontology#`, swoogle = `http://daml.umbc.edu/ontologies/webofbelief/1.4/swoogle.owl#`, daml = `http://www.daml.org/2001/03/daml+oil#`

Chapter 4

User Evaluation

This chapter presents the BigDataOcean Metadata Repository's user evaluation. It was divided in two questionnaires, a task scenario test in which users were asked to perform some activities and evaluate each task's complexity; and a usability test which was answered immediately after finishing the task scenario studies, to help determine whether the system is user-friendly and its ease of use.
It was important that both questionnaires were answered by the same user, this would give an overview of what the user thinks and how he/she feels about the system after performing the task scenarios test.

4.1 Participants

A total of eight participants evaluated the system, only seven evaluations were considered at the end. One participant did not understand there were two questionnaires, and only filled one, invalidating the answers of the filled questionnaire.
Only one participant is not related to the BigDataOcean project, the other seven participants are related to it, this gives more accurate results on how usable and easy to use is the system. Figure 4.1 shows the area of expertise, the degree of studies and job position of the participants.

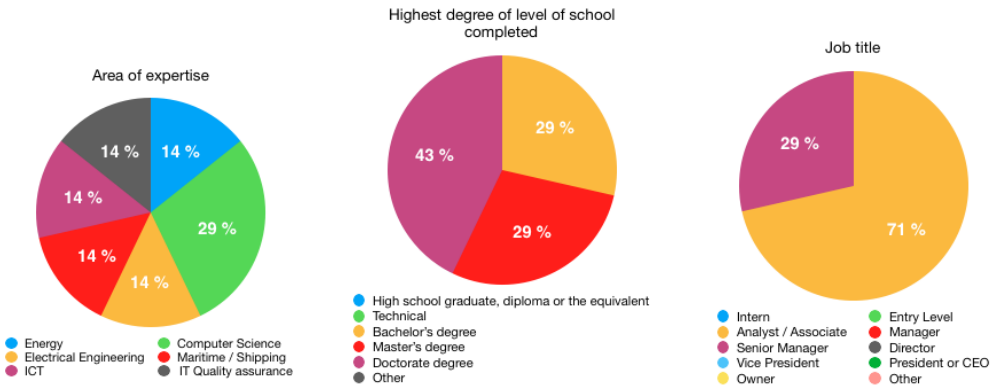


FIGURE 4.1: Demographics.

4.2 Task Scenarios Test

The task scenario test has six scenarios, each of them evaluates a functionality of BigDataOcean Metadata Repository. **Task 1** aimed to evaluate how users would find vocabularies and ontologies related to a particular pilot. **Task 2** goal was to search vocabularies/ontologies in the Metadata Repository based on different criteria and keywords. **Task 3** focused on how the user would find specific information about the described vocabularies/ontologies in the Metadata Repository. **Task 4** and **Task 5** evaluated how users would search for maritime-related and general terminology respectively, based on different criteria and keywords. **Task 6** aimed to evaluate SPARQL queries over the Metadata Repository and how users would use the provided SPARQL endpoint to access the metadata.
The goal of this test was to determine whether the users were able to navigate through the system and how difficult it was to find what they were asked.

For each of the tasks, users were asked to provide instructions on how they manage to complete the task, followed by four questions, and finally a section to provide further comments.
Question A asked difficulty of the task on a scale from 1 to 5, where 1 is Not hard, and 5 is Very hard, **question B** questions whether the information provided was useful on a scale from 1 to 5, where 1 is Not useful, and 5 is Very useful, **question C** asked if the participant was able to complete the task, **question D** requested to select the time the participant spent in the task, and **question E** asked to provide details of difficulties encountered while solving the task.

In the following subsections, for each task, it is explained what was asked, what was expected from the user and how the user performed in the task.

4.2.1 Task 1

One of the Big Data Ocean project pilots is about vessel's maritime security and anomaly detection. It focuses on keeping track of vessels' positions and irregular situations encountered as they travel across the seas, to do so, it is necessary to store their geographic location, what kind of situation they encountered, and more. Imagine your working group and that you are working on maintaining information about vessel's maritime security on the Mediterranean Sea. To describe this information you need specialized vocabularies. Walk me through the process of finding these vocabularies using the BDO Metadata Repository: `http://212.101.173.34:3333/dataset/bdo/`

4.2.1.1 Goal

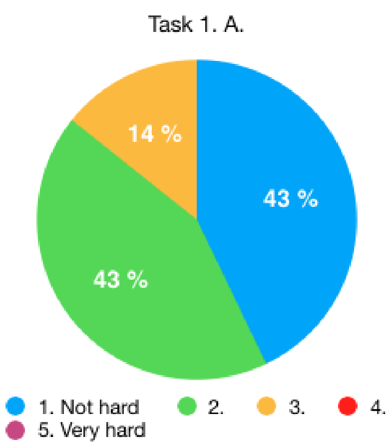
This task aimed to evaluate how users would find vocabularies and ontologies related to a certain pilot. Thus, it was expected that the user performed the following steps:

1. From the main page, choose PILOT tab,
2. select PILOT_III.

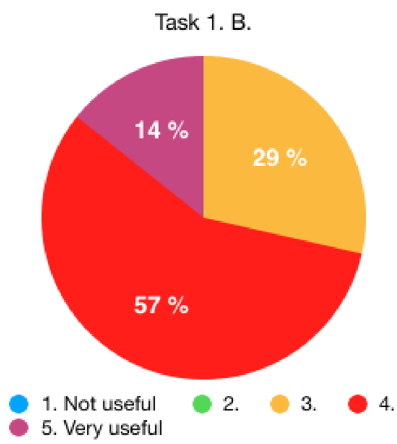
4.2.1.2 Answers

Six from seven participants completed the task by selecting the tab "PILOTS", next they chose "PILOT III" which is related to "Maritime Security". One participant used the "Search function" but he/she did not specify any search parameters entered.

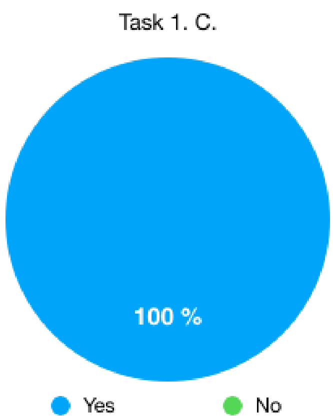
As shown in Figure 4.2 all participants were able to complete the task under 5 minutes, 57% thought the information provided was useful, and the majority of participants thought the task was not hard.



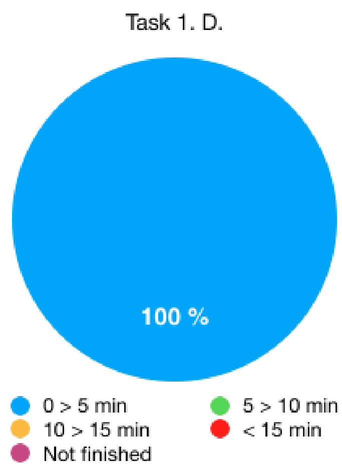
(A) How hard was to solve this task?



(B) Was the information provided useful?



(C) Were you able to complete the task?



(D) How long did it take to complete the task?

FIGURE 4.2: Task 1.

As for difficulties participants encounter, some participants did not understand how to find the required vocabularies; others were not sure if they needed to evaluate if the results were helpful or not.

4.2.2 Task 2

Now you are creating a new dataset that stores all sensor readings on the Mediterranean Sea. Please search for vocabularies/ontologies that are meant to describe this information and describe the process.

4.2.2.1 Goal

This task was to evaluate searching vocabularies/ontologies in the Metadata Repository based on different criteria and keywords. From the users was expected to follow these steps:

1. From the main page, choose VOCABS tab,
2. in the search field, type "sensor" or "mediterranean sea",
3. find vocabularies/ontologies that are related.

4.2.2.2 Answers

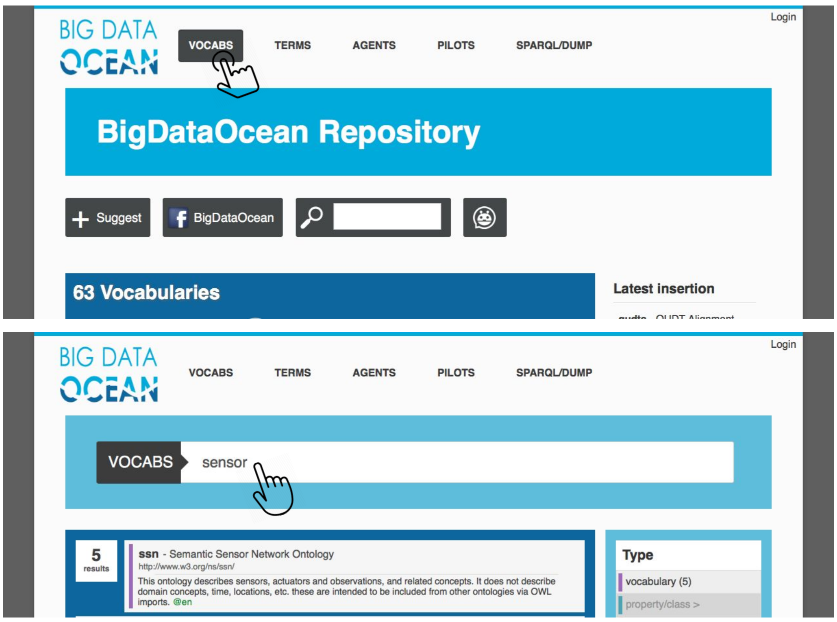


FIGURE 4.3: Steps that three participants performed in Task 2 of the task scenario test. Participants selected the tab "VOCABS" and used "sensor", "sensor reading" or "Mediterranean sea" as search parameters, receiving vocabularies and ontologies as result.

Three participants selected the tab "VOCABS" and used "sensor", "sensor reading" or "Mediterranean sea" as search parameters, receiving vocabularies and ontologies as a result. Two participants, instead of using the tab "VOCABS" chose the tab "TERMS" and searched for "sensor" or "location". One participant used the search function and found 26 results, once again, as in the previous task he/she did not provide the search parameters entered.

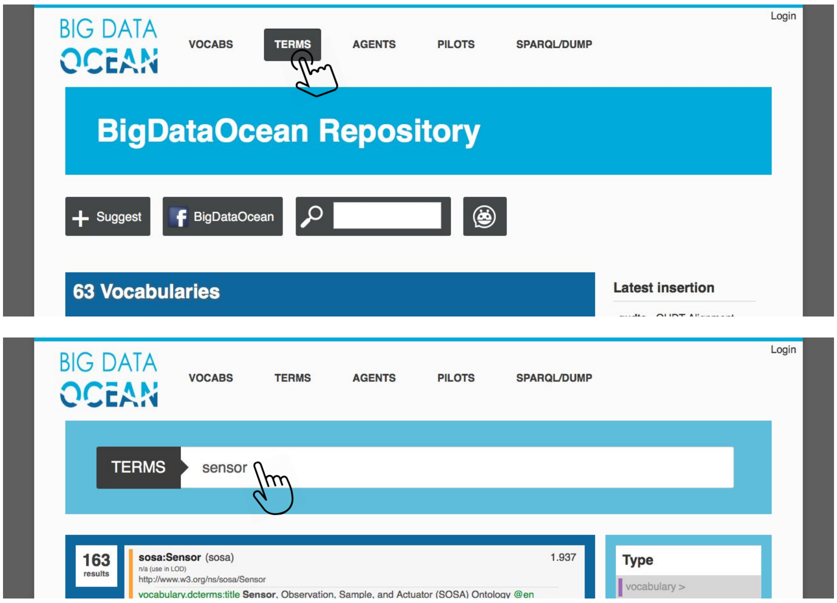


FIGURE 4.4: Steps that two participants performed in Task 2 of the task scenario test. Participants chose the tab "TERMS" and searched for "sensor" or "location", receiving vocabularies and ontologies as result.

All participants completed the task, even though one participant found the task hard and took him/her between 5 to 10 minutes to complete the task, the rest of participants finished before 5 minutes, see Figure 4.5.

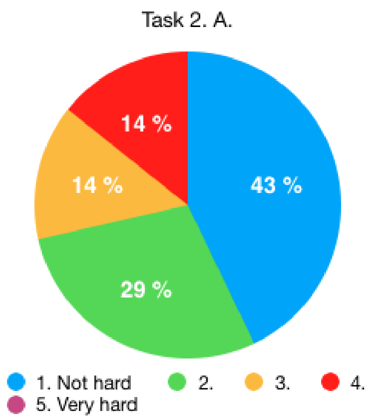
The difficulty encountered was understanding how to search for "Mediterranean Sea" related vocabularies.

4.2.3 Task 3

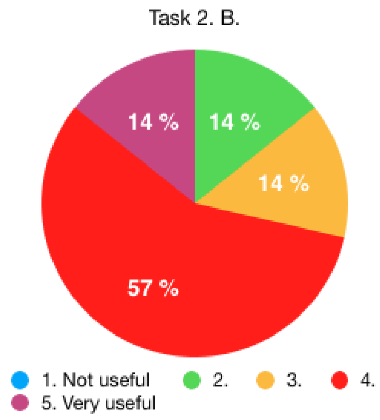
Now, you want to know, when one of the vocabularies found in the question before was last updated. Describe how would you find out this information.

4.2.3.1 Goal

This task is a continuation from Task 2 (4.2.2), it focused on how the user would find specific information about the described vocabularies/ontologies in the Metadata Repository. In this task, it was intended that the user

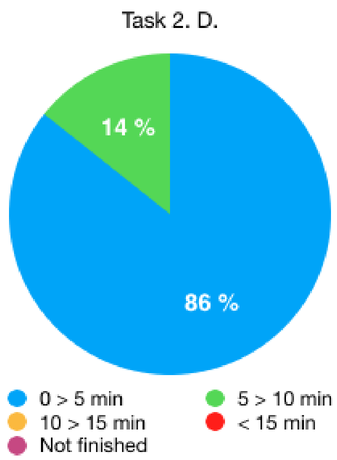


(A) How hard was to solve this task?



(B) Was the information provided useful?

(C) Were you able to complete the task?



(D) How long did it take to complete the task?

FIGURE 4.5: Task 2.

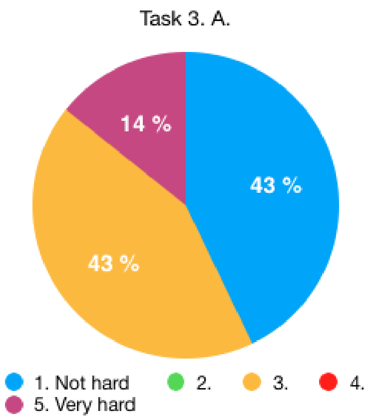
1. chose one vocabulary from the task before,
2. scroll down the displayed metadata until he/she find the section "Vocabulary Version History".

alternatively, the user clicked on the main page the "BDO Question Answering" logo, and asked "When was [vocab] last updated?".

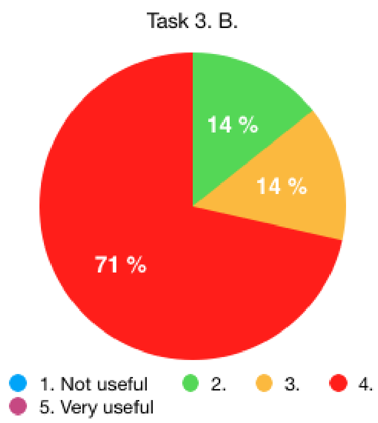
4.2.3.2 Answers

Four participants clicked on a vocabulary, three of them searched for the "Vocabulary Version History" section and found the date; the other participant did not give any details on how he/she found the last updated details. One participant, clicked on the main page "BDO Question and Answering" logo and asked, "*When was ssn last updated?*". Another participant did not provide any information except for "Vocabs → CF". Finally, the last participant wrote: "*URL section in the "Mediterranean Sea" page*". These two answers are confusing; they do not provide any insight on how participants completed the task.

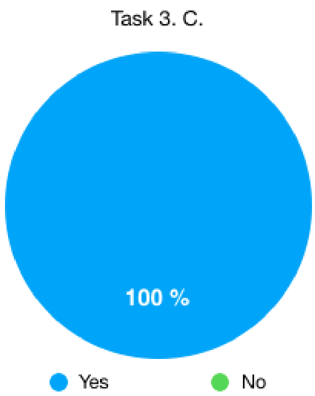
As shown in Figure 4.6, this task was solved by all participants, where the majority of participants (86%) finished under 5 minutes while 14% required between 5 and 10 minutes to finish the task. 71% of the participants thought the information provided was useful.



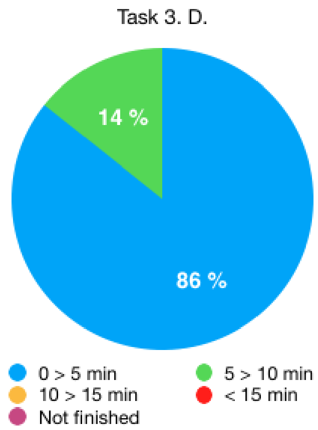
(A) How hard was to solve this task?



(B) Was the information provided useful?



(C) Were you able to complete the task?



(D) How long did it take to complete the task?

FIGURE 4.6: Task 3.

When asked about difficulties encountered, it seems like participants were not able to quickly find the last updated date, it was not clear for the participants, if the date under the label "Comment" was the last date updated, or if they had to search it by clicking on the URI of the vocabulary.

4.2.4 Task 4

You need specific terminology for describing the geographic location of each sensor. How would you do this?

4.2.4.1 Goal

The goal of this task was to evaluate how users would search for maritime-related terminology based on different criteria and keywords.
It was intended that the user perform one of the following steps:

- Select the Tag “Geography”, or
- from the main page, go to “TERMS” tab and search for “longitude” “latitude”...

4.2.4.2 Answers

One participant did not provide any information, another two participants wrote: "*same procedure as the previous task*" and "*similar to the previous ones*"; while four participants provided detailed information, two participants selected "TERMS" tab and searched for "sensor location" or "locations". One participant used the "geo" vocabulary and used its terms. Another participant clicked on the tag "Geography" from the "Category Tags" section on the main page, then selected "geo- WGS84 Geo Positioning".

Figure 4.7 displays the task was ambiguous, even if 86% of participants completed the task. How hard the task was, is not clear to say, 14% found it very hard, while the rest was between 1 Not hard and 3. 71% thought the provided information was useful, even though 14% said it was not useful. The task was completed by 71% of participants under 5 minutes, some participants finished between 5 to 10 minutes, and 10 to 15 minutes.

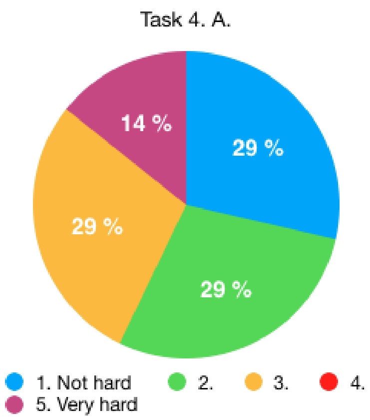
The difficulties of this task according to the participants were that they were not able to find what was requested and choosing the right terminology among many results.

4.2.5 Task 5

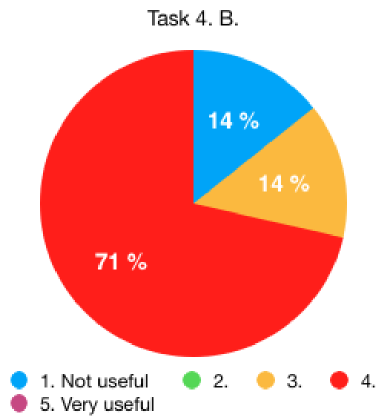
At the same time, you need to describe general data (e.g., title, description) about your dataset so that other people interested can read and understand what it is about. Please detail the steps you took to complete this task.

4.2.5.1 Goal

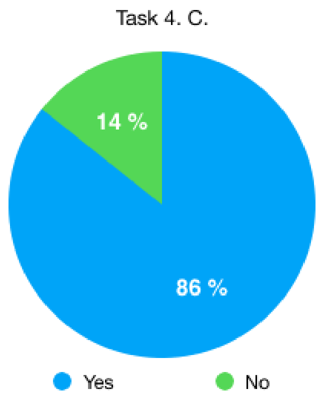
Task 5 goal was similar to Task 4, to evaluate how users would search for general terminology based on different criteria and keywords. General terminology refers to terminology to describe metadata, such as name, description, creation date, etc. Thus, it was expected that participants clicked on the main page in the TERMS tab and searched keywords like, “title” “description” “date”...



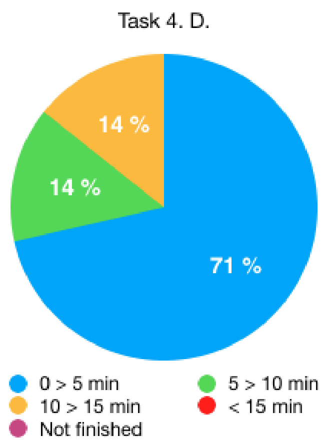
(A) How hard was to solve this task?



(B) Was the information provided useful?



(C) Were you able to complete the task?



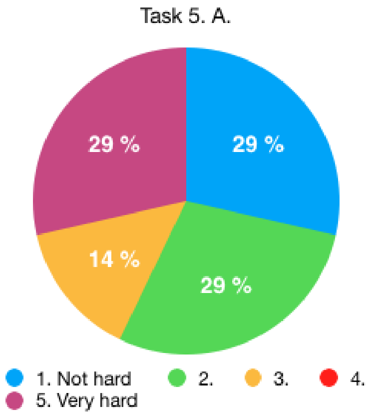
(D) How long did it take to complete the task?

FIGURE 4.7: Task 4.

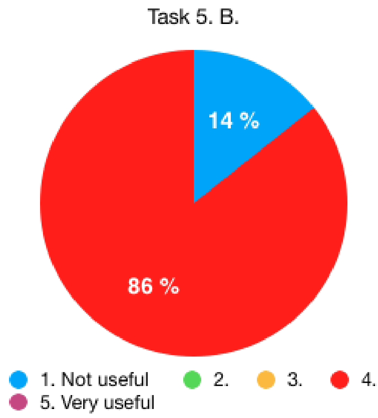
4.2.5.2 Answers

At this point seems like one participant got tired of writing the steps and instead wrote "*same as precious*". Other participant selected "TERMS" tab and "*search for each term*". Apparently, the task was not clear for a participant, he/she wrote "*Add metadata, tags, etc.?*", which does not give any insight on whether he/she search for these terms, or if he/she wanted to add this tags?. One other participant described two procedures, "*Main page → Category/Tags: "General & Upper" → "DUL" and "Schema" or "TERMS" tab → search: "title", "description",...*, which one other participant did too. Another participant searched for "*vocabulary "tags" you have "General" tab, so you get the "schema" vocabulary which should be useful to describe general data*".

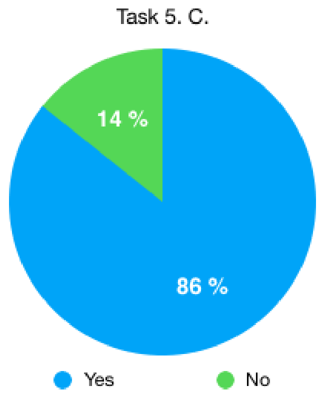
All participants did not complete this task, 86% completed the task while 14% did not; and in general the information provided was useful to complete the task, even though 29% of the participants required more than 5 to 10 minutes to complete the task, as shown in Figure 4.8.



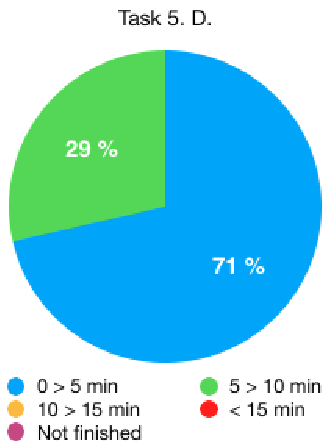
(A) How hard was to solve this task?



(B) Was the information provided useful?



(C) Were you able to complete the task?



(D) How long did it take to complete the task?

FIGURE 4.8: Task 5.

The primary difficulty of this task was choosing the correct one among many options.

4.2.6 Task 6

Imagine you want to perform a SPARQL query, to discover what vocabularies have been modified since 2014. Walk me through the process. (Hint: You can use Query Examples)

4.2.6.1 Goal

This task was designed to evaluate SPARQL queries over the Metadata Repository and how users would use the provided SPARQL endpoint to access the metadata. In order to do this, it was expected that the users followed the next steps:

1. Click on "SPARQL/DUMP" tab,

2. click on the drop-down list called "Query Examples",
3. choose the SPARQL query or write the query.

4.2.6.2 Answers

Six from seven participants were able to find the "SPARQL/DUMP" tab; one participant wrote: "*don't know SPARQL*". From the participants that found the tab, three participants found the section "Query Examples" and chose "Query #3" and executed the query with the "play" button. In contrast, one participant wrote "*write the query of interest, click on "play" button*". Also, two participants wrote they do not know SPARQL language.

Figure 4.9 demonstrates that 43% of the participants did not complete this task, one participant did not answer how long did it take to complete the task, and the rest required less than 5 minutes, or between 5 to 10 minutes to completed it. One participant did not select how useful was the information given. However, 66% chose useful, 4 or 5 Very useful in the scale, while 17% said it was not useful.

As mentioned before, the most significant difficulty was that participants do not know SPARQL language and they were not able to fully complete the task.

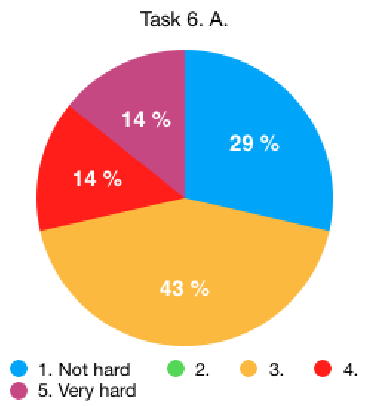
In general, all tasks were completed by all participants under 5 minutes, except for some cases that one or two participants were not able to complete the task or took them more than 5 minutes, but in all of the tasks, participants required less than 15 minutes to complete them. The difficulties encountered can be summarized by lack of technical knowledge, for example, in SPARQL language rules, or understanding the definition of terms in the context of BigDataOcean Metadata Repository.

4.3 Usability Test

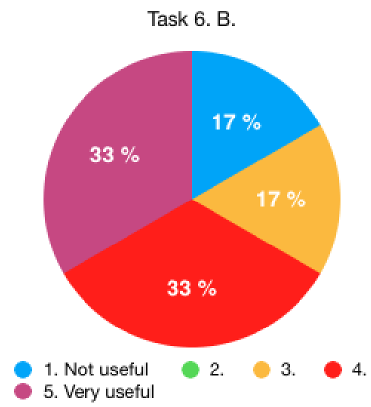
The usability test consisted of the System Usability Scale (SUS)[1] questions in order to determine the ease of use of BigDataOcean Metadata Repository after completing a series of tasks.
The SUS questionnaire consists of 10 statements, each of them with five answer options, from strongly disagree (1) to strongly agree (5) to measure the usability of hardware, software, etc.
The following are the statements the users were asked to evaluate from strongly disagree to strongly agree:

1. I think that I would like to use this system frequently.

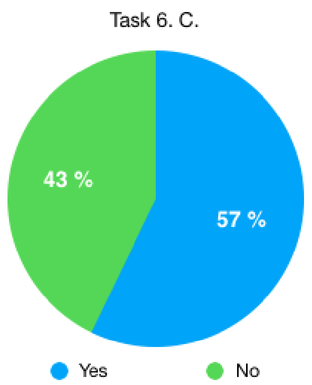
[1] https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html



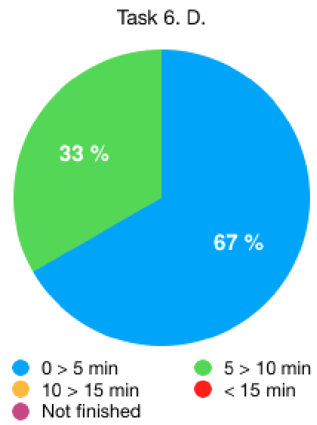
(A) How hard was to solve this task?



(B) Was the information provided useful?



(C) Were you able to complete the task?



(D) How long did it take to complete the task?

FIGURE 4.9: Task 6.

2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

Due to the low number of participants, it was not possible to measure the usability of the system according to SUS scores.

4.3.1 Goal

The goal of this test was to determine how easy to use is BigDataOcean metadata repository.

4.3.2 Answers

Figures 4.11, 4.12 show the results for each question, and Figure 4.10 depicts standard deviation for each of the questions.

In general, participants thought they would use the system, it was not complicated, they agreed the system was easy and not cumbersome to use, the system's functionalities were well integrated, and it does not have too many inconsistencies. They think people would learn quickly to use the system.

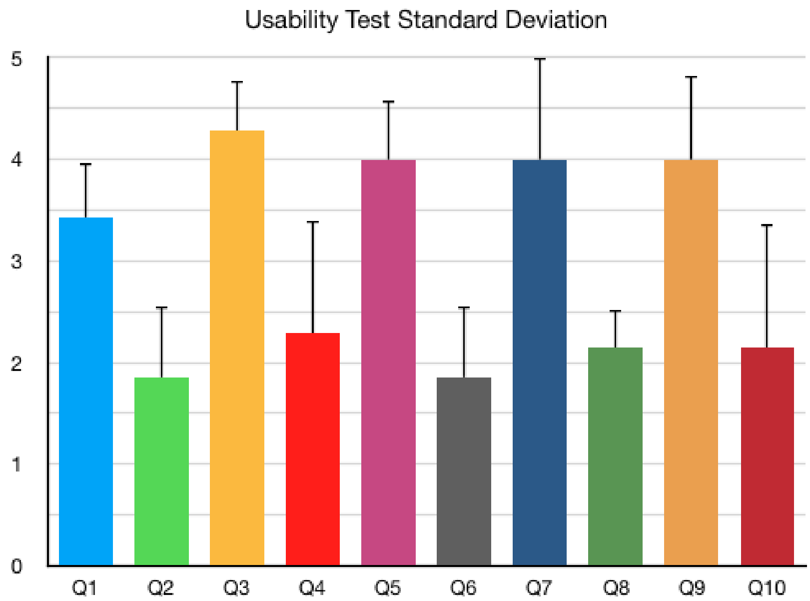


FIGURE 4.10: Usability test - Standard deviation per question

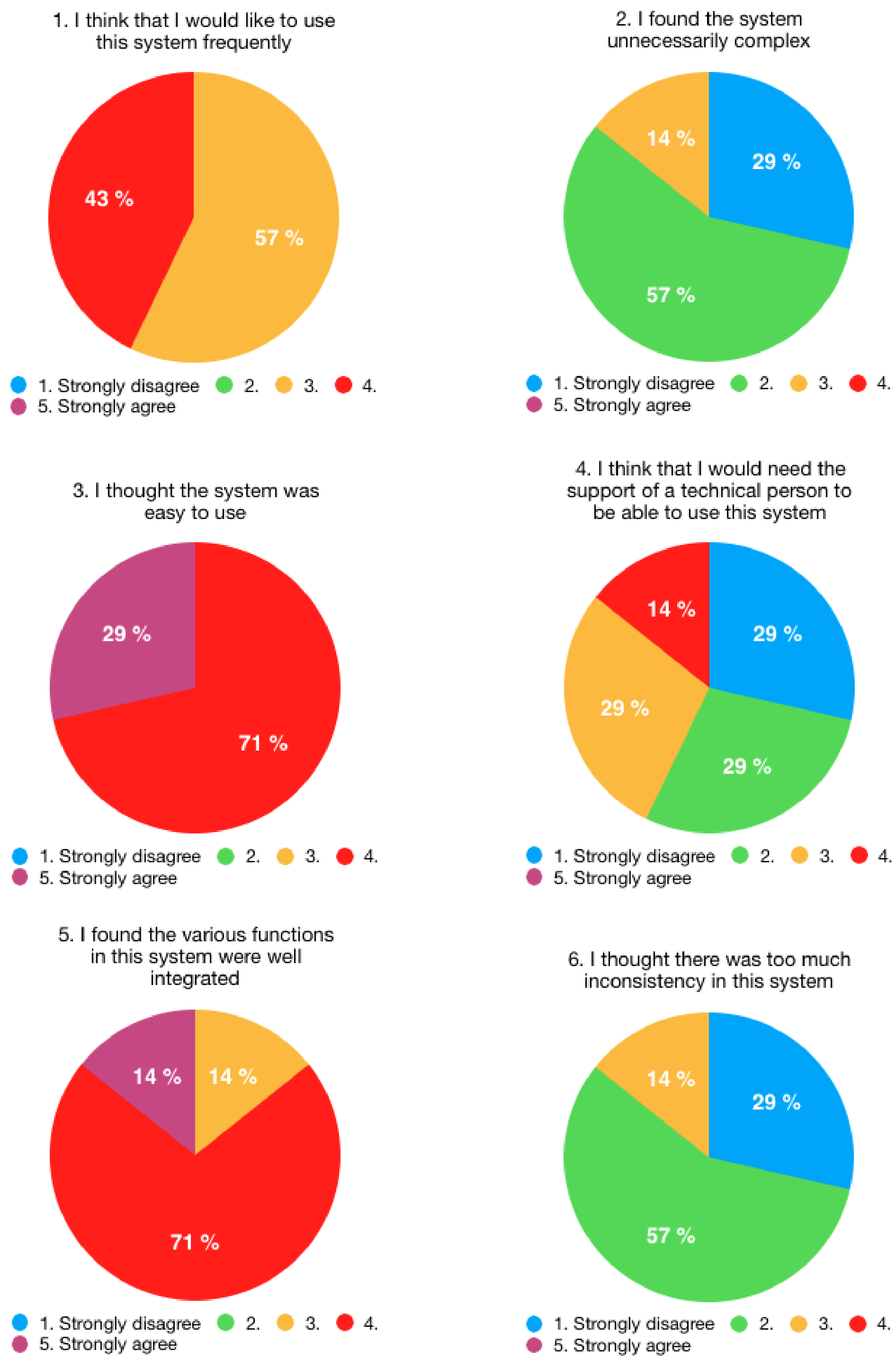


FIGURE 4.11: Usability Test, questions 1 to 6.

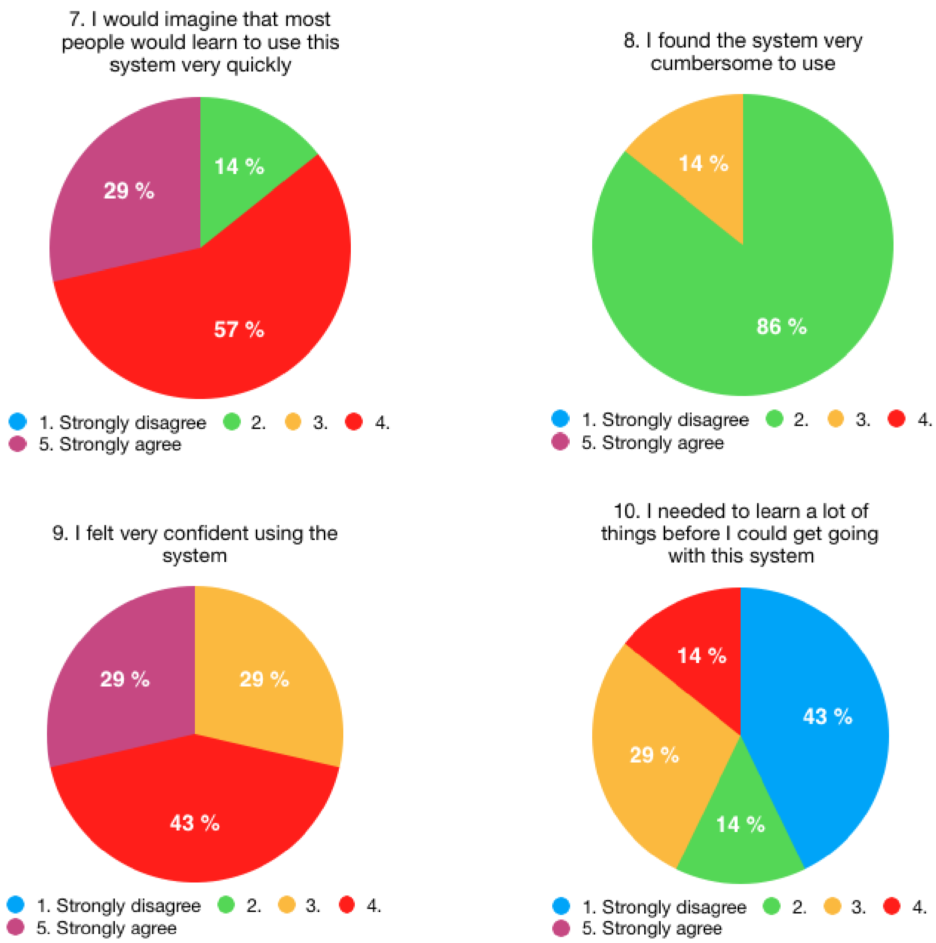


FIGURE 4.12: Usability Test, questions 7 to 10.

Chapter 5

Discussion

This section discusses the main findings from the literature survey and the user evaluation alongside with its limitations.

5.1 Findings from Literature Survey

Semantic Web is still a new topic and majorly studied and worked in the educational and research environment; that is why just a few publications are treating the topic of metadata repositories focusing on Semantic Web.

From these few publications, it is important to emphasize that six out of eight metadata repositories studied, are available online for users with a wide range of background, from experts, non-experts, and domain-related users. There has been an evolution in the features included in the metadata repositories since one of the firsts in 2003, till the most recent in 2017. From just letting users insert, modify and delete metadata about vocabularies and ontologies, to offering the ability to import metadata from published vocabularies. The earlier repositories allowed simple search over the metadata stored; now as a plus feature, they offer advanced search and the possibility to perform searches using the SPARQL Endpoint.

In the early systems, they did not store as much information as needed, now the latest repositories store all possible metadata from one vocabulary, and not only metadata about it but also statistics about the data of each of the vocabularies and ontologies.

As limitation some repositories did not offer a clear view about their data model, it was necessary to study the repository online, in order to try and identify what kind of metadata, which properties and vocabularies they are using. Additionally, some repositories were not available online anymore, which made the proper study of their features and data model difficult or impossible.

5.2 Findings from User Evaluation

From the user evaluation is possible to derive, users were able to perform all tasks without spending excessive time in the system; which means that the system is somewhat explicit about the relation between buttons names and their functionality. There were two cases in which the functionality of the buttons was not clear from the icons they displayed.

Having a more extensive set of participants, that are not related to the BigDataOcean project, would have given further insight on users that are not experts in the topic of Semantic Web. However, as mentioned before, Semantic Web is a topic that it is starting to reach companies and not only universities or research groups.

In general, BigDataOcean Metadata Repository is usable by experts and non-experts users, to perform some advanced search using the SPARQL Query Endpoint, users might need to have a certain level of knowledge in SPARQL query language or required the support of a technical person. The system is not complicated to use, and all its functions are well integrated.

Chapter 6

Conclusions

In this thesis, it was reported about the state of the art of Semantic Web metadata repositories, the case study BigDataOcean Metadata Repository and the methodology of how to decide whether to extend an existing metadata repository or to create a new one and the results of the evaluation of BigDataOcean Metadata Repository performed by users.

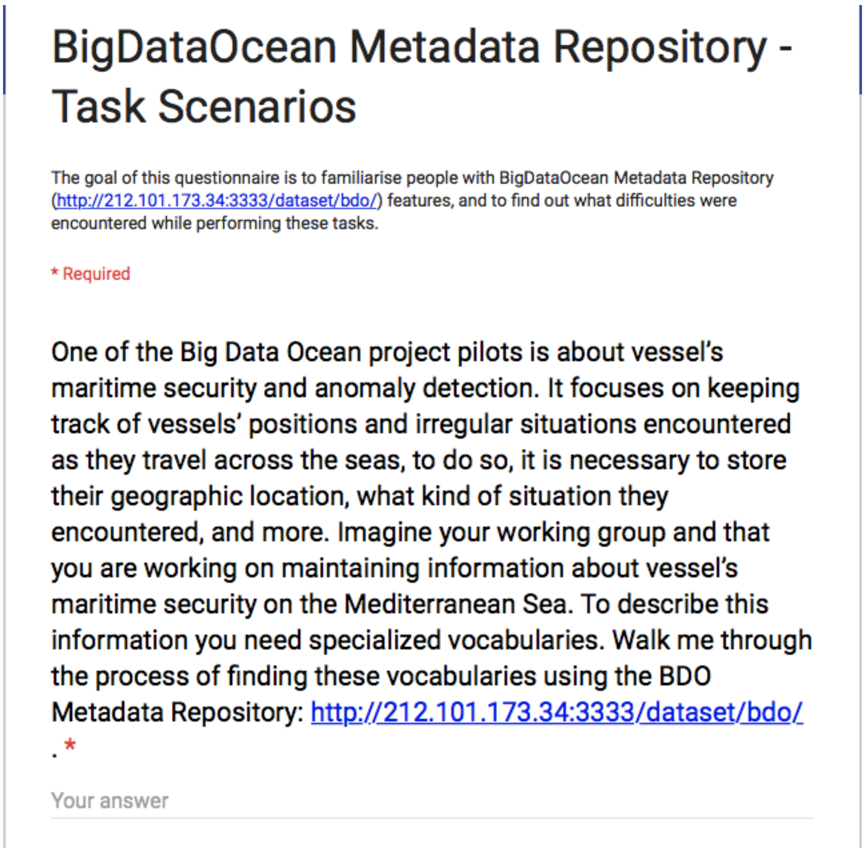
The majority of existing metadata repositories are not domain dependent, thus features dependent on a specific domain, are still missing. For example, for BigDataOcean Metadata Repository a functionality that connected stored metadata about ontologies and vocabularies, with each of the pilots from the project was missing; it was necessary to extend LOV in order to add the feature.
This experience taught us what steps should a developing team follow to decide if, for a specific project, it is better to create a new metadata repository or extend an already existing one.

BigDataOcean Metadata Repository has a medium scale architecture, it is published on the Web, but it is restricted to vocabularies and ontologies in the maritime domain. The number of vocabularies/ontologies for the maritime domain is quite small compared to the number of vocabularies required in the biomedical domain. Thus, using the BigDataOcean Metadata Repository architecture on a Web-scale would require extensive testing.

Until this point, BigDataOcean Metadata Repository is capable of extracting/importing metadata of already published on the Web vocabularies and ontologies, as future work, it is intended that the metadata repository allow extracting and importing metadata of locally saved ontologies and vocabularies. Also, to perform availability, response time and latency test of the BigDataOcean Metadata Repository, to improve those aspects in which the system fails according to the tests. Additionally, to perform a second user evaluation with a higher number of participants.

Appendix A

User Evaluation - Task Scenario Questionnaire



BigDataOcean Metadata Repository - Task Scenarios

The goal of this questionnaire is to familiarise people with BigDataOcean Metadata Repository (http://212.101.173.34:3333/dataset/bdo/) features, and to find out what difficulties were encountered while performing these tasks.

* Required

One of the Big Data Ocean project pilots is about vessel's maritime security and anomaly detection. It focuses on keeping track of vessels' positions and irregular situations encountered as they travel across the seas, to do so, it is necessary to store their geographic location, what kind of situation they encountered, and more. Imagine your working group and that you are working on maintaining information about vessel's maritime security on the Mediterranean Sea. To describe this information you need specialized vocabularies. Walk me through the process of finding these vocabularies using the BDO Metadata Repository: http://212.101.173.34:3333/dataset/bdo/ . *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

Task #2

Now you are creating a new dataset that stores all sensor readings on the Mediterranean Sea. Please search for vocabularies / ontologies that are meant to describe this information and describe the process. *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

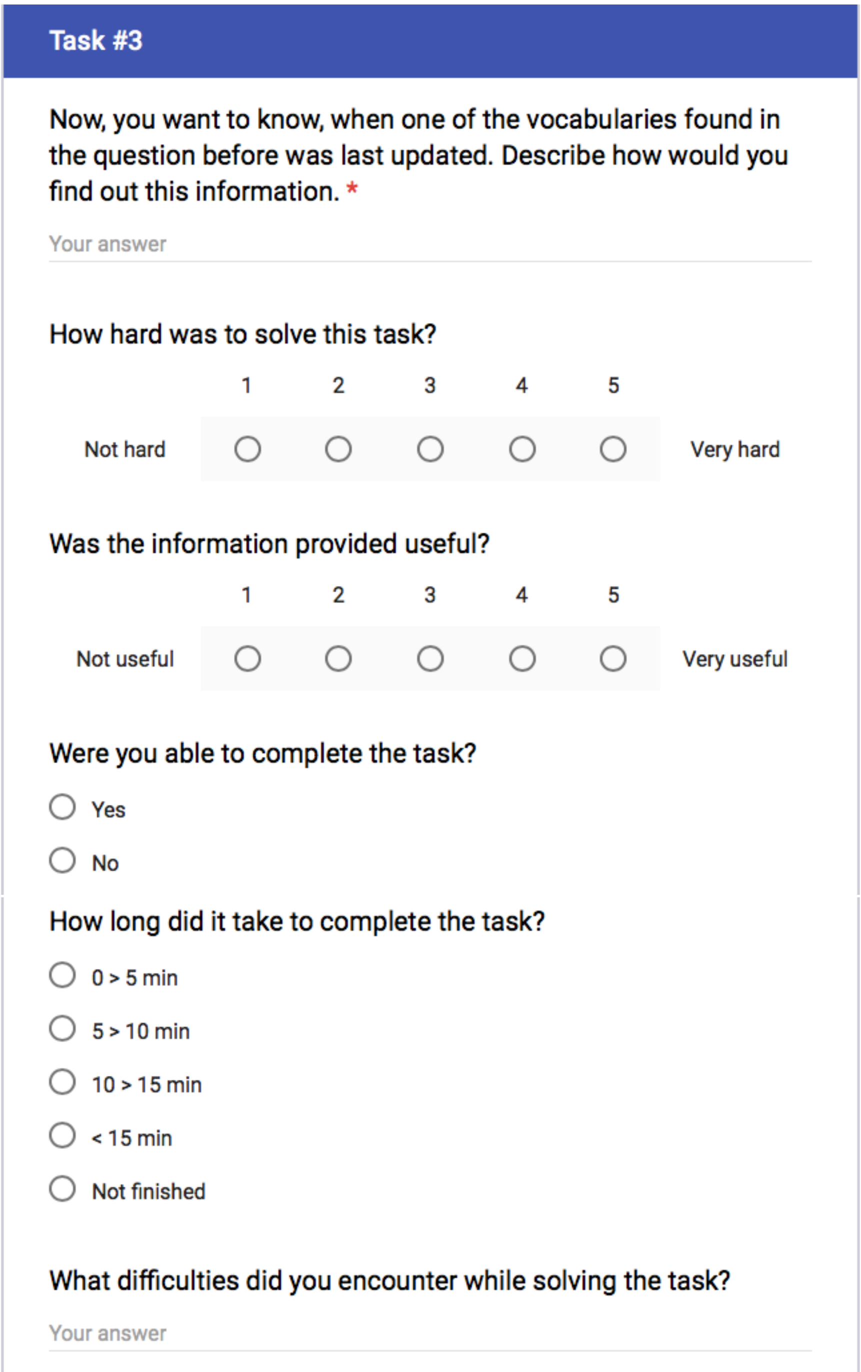
- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

What difficulties did you encounter while solving the task?

Your answer



Task #3

Now, you want to know, when one of the vocabularies found in the question before was last updated. Describe how would you find out this information. *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

What difficulties did you encounter while solving the task?

Your answer

Task #4

You need specific terminology for describing the geographic location of each sensor. How would you do this? *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

What difficulties did you encounter while solving the task?

Your answer

Task #5

At the same time, you need to describe general data (e.g., title, description) about your dataset so that other people interested can read and understand what it is about. Please detail the steps you took to complete this task. *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

What difficulties did you encounter while solving the task?

Your answer

Task #6

Imagine you want to perform a SPARQL query, to discover what vocabularies have been modified since 2014. Walk me through the process. (Hint: You can use Query Examples) *

Your answer

How hard was to solve this task?

	1	2	3	4	5	
Not hard	○	○	○	○	○	Very hard

Was the information provided useful?

	1	2	3	4	5	
Not useful	○	○	○	○	○	Very useful

Were you able to complete the task?

- ○ Yes
- ○ No

How long did it take to complete the task?

- ○ 0 > 5 min
- ○ 5 > 10 min
- ○ 10 > 15 min
- ○ < 15 min
- ○ Not finished

What difficulties did you encounter while solving the task?

Your answer

Demographics

What is your area of expertise?

Your answer

What is the highest degree or level of school you have completed?

- ○ High school graduate, diploma or the equivalent (for example: GED)
- ○ Technical
- ○ Bachelor's degree
- ○ Master's degree
- ○ Doctorate degree
- ○ Other:

Which of the following most closely matches your job title?

- ○ Intern
- ○ Entry Level
- ○ Analyst / Associate
- ○ Manager
- ○ Senior Manager
- ○ Director
- ○ Vice President
- ○ President or CEO
- ○ Owner
- ○ Other:

Appendix B

User Evaluation - Usability Questionnaire

BigDataOcean Metadata Repository - Usability Test

The goal of this test is to determine how easy to use is the metadata repository.

* Required

I think that I would like to use this system frequently *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the system unnecessarily complex *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I thought the system was easy to use *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I think that I would need the support of a technical person to be able to use this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the various functions in this system were well integrated *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I thought there was too much inconsistency in this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I would imagine that most people would learn to use this system very quickly *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the system very cumbersome to use *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I felt very confident using the system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I needed to learn a lot of things before I could get going with this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Demographics

What is your area of expertise? *

Your answer

What is the highest degree or level of school you have completed? *

- ○ High school graduate, diploma or the equivalent
- ○ Technical
- ○ Bachelor's degree
- ○ Master's degree
- ○ Doctorate degree
- ○ Other: ______

Which of the following most closely matches your job title? *

- ○ Intern
- ○ Entry Level
- ○ Analyst / Associate
- ○ Manager
- ○ Senior Manager
- ○ Director
- ○ Vice President
- ○ President or CEO
- ○ Owner
- ○ Other: ______

Bibliography

[1] Semantic web ontology. https://www.w3.org/standards/semanticweb/ontology, .

[2] Natalya F Noy, Nigam H Shah, Patricia L Whetzel, Benjamin Dai, Michael Dorf, Nicholas Griffith, Clement Jonquet, Daniel L Rubin, Margaret-Anne Storey, Christopher G Chute, et al. Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, 37(suppl_2):W170–W173, 2009.

[3] Carlos Rueda, Luis Bermudez, and Janet Fredericks. The mmi ontology registry and repository: A portal for marine metadata interoperability. In *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, pages 1–6. IEEE, 2009.

[4] Till Mossakowski, Oliver Kutz, and Mihai Codescu. Ontohub: A semantic repository for heterogeneous ontologies. In *IN: PROC. OF DACS'14 (2014*. Citeseer, 2014.

[5] Pierre-Yves Vandenbussche, Ghislain A Atemezing, María Poveda-Villalón, and Bernard Vatant. Linked open vocabularies (lov): a gateway to reusable semantic vocabularies on the web. *Semantic Web*, 8(3): 437–452, 2017.

[6] Quddus Chong, Anup Marwadi, Kaustubh Supekar, and Yugyung Lee. Ontology based metadata management in medical domains. *Journal of Research and Practice in Information Technology*, 35(2):139–154, 2003.

[7] Li Ding, Tim Finin, Anupam Joshi, Rong Pan, R Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi, and Joel Sachs. Swoogle: a search and metadata engine for the semantic web. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 652–659. ACM, 2004.

[8] Swoogle. http://swoogle.umbc.edu/2006/.

[9] Raúl Palma, Peter Haase, and Asunción Gómez-Pérez. Oyster: sharing and re-using ontologies in a peer-to-peer community. In *Proceedings of the 15th international conference on World Wide Web*, pages 1009–1010. ACM, 2006.

[10] Bioportal. https://bioportal.bioontology.org.

[11] Mathieu d'Aquin, Jérôme Euzenat, Chan Le Duc, and Holger Lewen. Sharing and reusing aligned ontologies with cupboard. In *Proceedings of the fifth international conference on Knowledge capture*, pages 179–180. ACM, 2009.

[12] Marine metadata interoperabilty ontology registry and repository. http://mmisw.org/ont/#/.

[13] Ontohub. https://ontohub.org, .

[14] Linked open vocabularies (lov). http://lov.okfn.org/dataset/lov/.

[15] Paul Buitelaar, Thomas Eigner, and Thierry Declerck. Ontoselect: A dynamic ontology library with support for ontology selection. In *In Proceedings of the Demo Session at the International Semantic Web Conference*. Citeseer, 2004.

[16] Eero Hyvönen, Kim Viljanen, and Osma Suominen. Healthfinland—finnish health information on the semantic web. In *The Semantic Web*, pages 778–791. Springer, 2007.

[17] Eero Hyvönen, Miikka Junnila, Suvi Kettula, Eetu Mäkelä, Samppa Saarela, Mirva Salminen, Ahti Syreeni, Arttu Valo, and Kim Viljanen. Publishing museum collections on the semantic web: the museumfinland portal. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 418–419. ACM, 2004.

[18] Mathieu d'Aquin, Marta Sabou, Martin Dzbor, Claudio Baldassarre, Laurian Gridinoc, Sofia Angeletou, and Enrico Motta. Watson: A gateway for the semantic web. 2007.

[19] Rdf schema 1.1. https://www.w3.org/TR/rdf-schema/, .

[20] Owl web ontology language guide. https://www.w3.org/TR/owl-guide/.

[21] Skos simple knowledge organization system reference. https://www.w3.org/TR/skos-reference/.

[22] Rdf 1.1 xml syntax. https://www.w3.org/TR/rdf-syntax-grammar/, .

[23] Rdf 1.1 turtle. https://www.w3.org/TR/turtle/.

[24] Daml+oil (march 2001) reference description. https://www.w3.org/TR/daml+oil-reference#2.

[25] Dublin core metadata element set, version 1.1: Reference description. http://dublincore.org/documents/dces/.

[26] Rdf 1.1 concepts and abstract syntax. https://www.w3.org/TR/rdf11-concepts/, .

[27] Juha Hakala. Dublin core metadata initiative. *CRIS 2000-Current Research Information Systems*, 2000.

[28] Ioanna Lytra, Fabrizio Orlandi, and Maria-Esther Vidal. BigDataOcean - Exploiting Oceans of Data for Maritime Applications, May 2017. URL https://doi.org/10.5281/zenodo.815343. https://2017.eswc-conferences.org/program/eu- project-networking-session.

[29] Bigdataocean deliverable, d3.1 bigdataocean linked data vocabularies and metadata repository architecture. http://www.bigdataocean.eu/site/wp-content/uploads/2016/12/BigDataOcean_Linked_Data_Vocabularies_and_Metadata_Repository_Architecture_v1.00.pdf,.

[30] Bigdataocean deliverable, d2.1 analysis report on big data components, tools and methodologies. http://www.bigdataocean.eu/site/wp-content/uploads/2017/04/BigDataOcean_Analysis_Report_on_Big_Data_Components_Tools_and_Methodologies-v1.00.pdf,.

[31] Roger S Pressman. *Software engineering: a practitioner's approach*. Palgrave Macmillan, 2005.