

LINKING FISH SPECIES DATA

**A DISSERTATION SUBMITTED TO THE UNIVERSITY OF
MANCHESTER FOR THE DEGREE OF MASTER OF SCIENCE IN
THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES**

2011

GERARDO ALBERTO DELGADO MONROY

SCHOOL OF COMPUTER SCIENCE

Table of Contents

ABSTRACT.....	5
DECLARATION.....	6
INTELLECTUAL PROPERTY STATEMENT.....	7
ACKNOWLEDGEMENTS.....	8
1. INTRODUCTION.....	9
2. BACKGROUND.....	12
2.1. Linked Data Overview.....	12
2.2. Linked Data Supporting Technologies.....	14
2.2.1. RDF and URIs.....	14
2.2.2. RDF Schema.....	18
2.2.3. OWL.....	18
2.2.4. SKOS.....	19
2.2.5. SPARQL.....	20
2.3. Integration of Datasets.....	22
2.4. The Web of Data.....	26
2.5. The FishDelish Project.....	27
2.6. The FishLink Project.....	28
2.7. Other Resources of Species on the Web.....	29
2.8. Biological Background Information.....	32
2.8.1. Taxonomic Classification of Fish.....	32
2.8.2. Scientific Names.....	35
2.9. URIs in the Biodiversity Domain.....	36
3. IMPLEMENTATION.....	38
3.1. Dataset Comparison Application.....	39
3.2. Identification of Prospective Datasets.....	41
3.2.1. Use of Linked Data Catalogues.....	42
3.2.2. Use of Linked Data Search Engines.....	44
3.2.3. Use of Linked Data Browsers.....	46
3.2.4. Analysis of Results.....	47
3.3. Analysis and Selection of Datasets.....	48
3.3.1. Methodology.....	48
3.3.2. BBC Wildlife Finder.....	50
3.3.3. Uniprot Taxonomy.....	54
3.3.4. EUNIS (European Nature Information System).....	56
3.3.5. Geospecies.....	59
3.3.6. Fishes of Texas.....	62
3.3.7. Spire.....	62
3.3.8. Taxonconcept.....	63
3.3.9. DBpedia.....	65
3.3.10. Freebase.....	69
3.3.11. OpenCyc.....	72
3.3.12. Bio2RDF.....	76
3.3.13. Global Names Index.....	80
3.3.14. Decision matrix construction.....	81
3.3.15. Analysis of results.....	83
3.4. Identification and Selection of Predicates.....	84
3.4.1. Identification of Vocabularies and Predicates.....	85
3.4.1.1. MuSim, the Similarity Ontology.....	86
3.4.1.2. Biological Taxonomy Vocabulary.....	86

3.4.1.3. Taxonconcept Ontology	86
3.4.1.4. The Association Ontology	87
3.4.1.5. SKOS	87
3.4.2. Selection of the Predicates.....	87
3.5. Construction of the RDF Triples.....	90
3.5.1. Preprocessing of the Authority Strings.....	92
3.5.2. Approximate Matching of the Authority Strings.....	94
3.5.3. Algorithm Used for the Construction of the Links.....	97
3.5.4. Number of Links Created.....	98
3.6. The Fish Species Finder Application.....	100
4. CONCLUSIONS.....	106
REFERENCES.....	110
BIBLIOGRAPHY.....	115
APPENDICES.....	117
Appendix A – The Linking Open Data cloud.....	117

Word count: 34079

List of Figures

Figure 1: RDF graph example.....	16
Figure 2: Relationship among the most popular species repositories.....	31
Figure 3: The taxonomic classification system.....	32
Figure 4: Classification of fish species	34
Figure 5: Classification of fish species 2.....	34
Figure 6: Dataset Comparison Application class diagram.....	40
Figure 7: BBC Wildlife Finder number of entities.....	52
Figure 8: EUNIS number of fish species entities.....	58
Figure 9: Phylum chordata in Geospecies.....	61
Figure 10: Spire search interface	63
Figure 11: Freebase number of animal entities.....	72
Figure 12: OpenCyc application execution.....	73
Figure 13: OpenCyc application interface.....	75
Figure 14: General decision matrix	82
Figure 15: Standard and extended link graph.....	89
Figure 16: Proportion of links containing skos:exactMatch and skos:closeMatch.....	99
Figure 17: Proportion of links connecting FishDelish with the other datasets.....	99
Figure 18: Fish Species Finder.....	101
Figure 19: Fish Species Finder search.....	101
Figure 20: Fish Species Finder, Betta splendens.....	102
Figure 21: Fish Species Finder, authority comparison.....	103
Figure 22: Fish Species Finder, Arripis trutta Taxonconcept data	104

List of Tables

Table 1: List of datasets found with the use of search engines.....	45
Table 2: Summary of the datasets found with each mechanism	47
Table 3: BBC Wildlife Finder data accessibility.....	51
Table 4: BBC Wildlife Finder estimated number of triples per species.....	53
Table 5: Uniprot data accessibility.....	55
Table 6: Uniprot estimated number of triples per species.....	56
Table 7: EUNIS data accessibility.....	57
Table 8: EUNIS estimated number of triples per species.....	58
Table 9: Geospecies data accessibility.....	60
Table 10: Geospecies estimated number of triples per species.....	62
Table 11: Taxonconcept data accessibility.....	64
Table 12: Taxonconcept estimated number of triples per species.....	65
Table 13: DBpedia data accessibility.....	67
Table 14: Dbpedia estimated number of triples per species.....	69
Table 15: Freebase data accessibility.....	71
Table 16: OpenCyc data accessibility.....	74
Table 17: Bio2RDF estimated number of triples per species.....	79
Table 18: Decision Matrix.....	83
Table 19: Estimation of the popularity of prospective predicates.....	88
Table 20: Number of authority matches before and after the string preprocessing.....	93
Table 21: Examples of Levenshtein distances less than four.....	96
Table 22: Examples of Levenshtein distances larger than three.....	96
Table 23: Predicates used for the construction of links.....	97
Table 24: Number of RDF links generated.....	98

ABSTRACT

The traditional World Wide Web (WWW) is built upon a set of documents which are interconnected through the use of hypertext links. Those documents are written in HTML, a language used to layout information for human consumption. However, web browsers and other software tools are not aware of the data contained in HTML documents, which reduces the possibility to processing that data.

Linked Data is a relatively new set of principles to share information on the WWW, where not only are documents published and interconnected, but also the data they contain, creating new possibilities for the processing, integration and discovery of information by both humans and applications. With the use of Linked Data, the link that joins two entities is typed, providing information about the relationship between those two elements.

FishDelish is a Linked Data project that exposes a large portion of FishBase, one of the most popular and comprehensive fish species databases available on the Web, as Linked Data.

The main purpose of this dissertation is the generation of RDF links to connect fish species data contained in the FishDelish dataset to other resources available in the Linked Data cloud, and expose some of the benefits of those links with the development of an application; trying to find practical and effective solutions to the challenges that represent this integration.

DECLARATION

I declare that no portion of the work referred to in this dissertation has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

INTELLECTUAL PROPERTY STATEMENT

- i. The author of this dissertation (including any appendices and/or schedules to this dissertation) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.
- ii. Copies of this dissertation, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has entered into. This page must form part of any such copies made.
- iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the dissertation, for example graphs and tables (“Reproductions”), which may be described in this dissertation, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.
- iv. Further information on the conditions under which disclosure, publication and commercialisation of this dissertation, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see <http://documents.manchester.ac.uk/display.aspx?DocID=487>), in any relevant Dissertation restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.manchester.ac.uk/library/aboutus/regulations>) and in The University’s Guidance for the Presentation of Dissertations.

ACKNOWLEDGEMENTS

I would like to thank sincerely my supervisor, Sean Bechhofer, for his excellent guidance, support and advice in the development of this project.

I would also like to thank my parents for the encouragement, support and motivation they have provided to me throughout my entire life.

1. INTRODUCTION

In 2006, Tim Berners-Lee [1] outlined the following four principles of Linked Data:

1. Use Uniform Resource Identifiers (URIs) as names of things.
2. Use HTTP URIs to allow people to look up those names.
3. Provide useful data using standardised technologies when a URI is looked up.
4. Include links to other URIs.

During the last few years a large number of datasets have been published in RDF format following these principles, which has led to the generation of a distributed repository of data containing millions of assertions about a wide range of topics. This data space is known as the Web of Data, and offers new possibilities for the integration of information from multiple resources published in a format that can be manipulated easily by applications [2].

The fourth principle specifies the creation of links to other Linked Data resources, with the purpose of increasing the ability to discover further data by connecting information from different resources. As Heat [3] suggests, data from different providers distributed across the Web can be integrated with a reduction in the technical difficulties that represent the connection of heterogeneous data. This allows the aggregation of several sources of information to offer a wider view of the domain of interest.

FishDelish is a dataset from the biological domain that has been created according to the principles of Linked Data and contains data on approximately 32,000 fish species. However, this data is not linked to other resources of fish species information with the use of semantic technologies. Therefore, the generation of links to other data sources published as Linked Data contributes to achieve one of the four basic principles for the construction of the Semantic Web. Moreover, those links will provide a more complete view about fish species to the users of FishDelish.

A number of resources containing information about fish species are available in the Web of Data. However, the selection of the most adequate datasets for the creation of the link requires a comprehensive analysis of them. Characteristics, such as the amount and type of information that they contain and the accessibility to their data, are some of the elements that must be taken into account.

Moreover, the integration of species data represents some challenges inherent to the biodiversity domain. The identification of entities representing the same species from different datasets requires an understanding of the nomenclature of species and their taxonomic classification.

Thus, the main purpose of this project is to integrate the FishDelish dataset with other Linked Data resources containing information of fish species, and develop an application to exemplify how these links can be used to combine information from different resources to present a comprehensive view of the topic. The integration is produced with the construction of RDF links connecting fish species entities. In order to perform this integration, a number of tasks must be performed:

- Find relevant datasets in the Linked Data cloud that contain information of the domain of interest; fish species data.
- Analyse the different datasets that are found, and select the most adequate and comprehensive resources of information.
- Design and implement a procedure to map the elements in the FishDelish dataset with the corresponding elements in the target datasets, taking into account some biological concepts for the recognition of entities referring to the same species.
- Select the most appropriate predicates, in terms of semantics, to construct the links.
- Build software to perform the construction of the links in an automated way.

According to Bizer et al. [4], the generation of the links between the FishDelish dataset and the other datasets, allows applications and humans to navigate between different data sources and discover further data and, simultaneously, the FishDelish data source is more integrated in the Web of Data.

The application is provided to offer a demonstration of how data from different resources can be combined with the use of RDF links to show a wider perspective of the topic of interest. In particular, this application allows the search of fish species' names to present a set of data taken from the different resources, where the user can compare the information from different data sources in a single interface. This design was motivated by the fact that basic information about species sometimes can differ from one resource to another. Thus, this tool allows the identification of disagreements in the information published by different data providers.

A detailed description of the procedure applied for the generation of links is provided, discussing the technical difficulties encountered and how they were addressed. Finally, an analysis of the

methodology applied is presented, discussing its advantages and limitations, and how this procedure can be applied in other circumstances.

The deliverables of the project are a collection of links contained in RDF files in the form of triples, and the application created to exemplify the benefits of the aggregation of data using Linked Data technologies.

A number of technologies are required for the development of this project, including schema languages to define, organise and link data in machine-readable format (i.e. RDF, OWL, SKOS) and a query language to retrieve RDF data (SPARQL). An overview of those technologies is offered in the background section. Additionally, some biological aspects are discussed to understand some of the issues that arise with the integration of information from the biological domain.

2. BACKGROUND

2.1. *Linked Data Overview*

The World Wide Web (WWW) is built upon a set of documents containing text, images and other media resources. However, the data contained in these documents is embedded in the plain text of the HTML files, with no standardised mechanisms to separate, identify and extract, with automated tools, the relevant data contained in web pages; raw data and presentation mark-up are mixed in the HTML files. This set of documents is known as the Web of Documents.

In this scenario, the user must extract data manually from the web pages, without the possibility of using standardised tools to perform automatically the extraction of the required data. For example, if a user accesses a web page about the city of Manchester and wants to know its current population, he or she has to read the article to find the part of the text where this data is mentioned, or use the search functionality of the browser to find the word “population”. However, this last approach could give more than one result because the word may occur several times under different contexts, and the user must identify which is the data that corresponds to the current population of all the found matches of the string. Again, this is a manual procedure performed by the user.

A software script could be created to extract the number associated with the string “population” in a web page. Nevertheless, this data extraction would not be reliable, because if that word is contained in the text more than once, for example, referring to the population in a past year, the foreign population or the population of the whole Greater Manchester, the software would not have a mechanisms to identify reliably the string that corresponds to the current population of the city.

Hence, we say that the information published in the WWW in the standard format is human readable, which means that the information is intended for human consumption and a computer program hardly can extract data from this kind of resources. The Web was developed as a space to share content to be read by humans rather than for data that can be processed meaningfully by computer programs [5].

To combat this problem, in 2006 Tim Berners Lee published a set of practices for publishing and linking data on the Web [1], for the creation of a global space where not only documents are published and connected but also the data, and is understandable not only by humans but also by

applications. These practices are known as the Linked Data principles.

Bizer et al. [4] report that during the last three years a great number of data providers have adopted the principles for publishing and linking data on the Web as Linked Data, which has enabled the creation of a global dataset called the Web of Data or Semantic Web. It is important to point out that Semantic Web is a broader term encompassing further functionalities and components, such as support for logical reasoning. However, many authors use both terms interchangeably.

Heath [3] comments that the Web of Data is not a separate entity different from the Web of Documents; rather it is an additional layer highly interlinked with the Web of Documents. Therefore, the Web of Data is not intended to replace the traditional Web, but to extend it by adding a layer containing data structured with standardised mechanisms and named links to other machine-readable resources. Thus, the Web of Data complements the standard Web, as they serve different purposes. In brief, the Linked Data technology is intended to publish and link in the WWW datasets in machine-readable format.

At this point, the question arises of the benefits of publishing and linking information as Linked Data. The answer is not simple, because this is a relatively new set of practices that have been applied in a more extensive manner during the past few years, and the full potential of this technology probably has not been discovered. What can be said safely is that Linked Data opens new possibilities for the processing and exchange of data in the Web, easing the development of domain specific applications and other tools.

Currently, some of the most common examples of Linked Data applications are mashups that combine data from different resources to provide a richer interface and comprehensive results to the user. One of the most popular examples of a mashup is Dbpedia Mobile. Intended for tourists exploring cities, it runs on mobile devices and provides a map with information of nearby places based on GPS positioning. This enables the navigation into different interlinked datasets to offer a rich experience to the user, who can explore his or her surroundings by reading related information and looking at photographs of such places [6].

Before the advent of Linked Data, the available means to produce applications by combining different Web resources (mashups) were Web APIs (Application Programmable Interfaces), which provide a mechanism to programmatically access data from web data sources via the HTTP protocol. Heath & Bizer [2] recognise that even when the benefits of using Web APIs to access

structured data in the Web are undeniable, the following disadvantages for the creation of mashups exist. Firstly, the programmer must know the methods for accessing data from every single API; secondly, the developed application contains highly specialised code to access each resource. More disadvantages of Web APIs are discussed widely in the related literature [2] [4].

Linked Data applications provide a further advantage over Web APIs mashups: according to Bizer [7], they can discover new resources at runtime, enabling the generation of more comprehensive results.

One more benefit of applying Linked Data techniques to publish and link structured data in the Web is the encouragement of the re-usability of data. Heath & Bizer [2] claim that if the structure of the data is standardised, the creation of applications to process and reuse it is easier. Interestingly, as [3] suggests, a dataset published in the Web of Data can be reused and presented in a way that was not anticipated by the provider, as data and presentation are separated in this architecture, allowing the creation of independent presentation layers.

In conclusion, Linked Data reduces the technical difficulties that represent the integration of multiple data sources while increasing the re-usability of data. This enables the development of a new generation of applications that take advantage of the machine-readable format, while offering new mechanisms for navigating, querying and extracting information from the WWW by using the current infrastructure of the standard Web and standardised technologies, creating a global data space of structured data.

2.2. Linked Data Supporting Technologies

2.2.1. RDF and URIs

Linked Data uses a number of standardised technologies. One of the most important is the Resource Description Framework (RDF), which is a language for representing information about things to be processed by computer programs [8]. RDF is a W3C (World Wide Web Consortium) specification which is based on two main principles: Identifying things of the real world by using Uniform Resource Identifiers (URIs), and making statements about things with the use of triples.

The first principle indicates that a URI, a short string that identifies a resource in the WWW, is assigned to every entity represented in the Web, allowing the unique identification of each element. It

can be compared to a primary key in a relational database, but the scope of a public URI is the whole WWW. URIs can be assigned to any kind of abstract or physical entity, such as people, songs, images, fish species and proteins. This practice guarantees that the information contained in a RDF file is not just text, but concepts associated with a definition that can be found by computer programs and humans in the Web [5].

It is worth mentioning that a URI is the generalisation of the concept of URL (Uniform Resource Locator). However, as Manola et al. [8] explain, URLs are intended to identify things that have network locations (IP addresses), while URIs can be used to refer to any type of element in the web, including:

- Network accessible resources (files, images, web services, HTML documents)
- Non-network accessible things (companies, people, fish species)
- Abstract concepts (songs, taxonomic groups).

An RDF document is comprised of a set of triples that embody statements about the entity being described. A triple consists of a subject, a predicate and an object, and can be represented as a graph with nodes and arcs. In general, the three elements of a triple are URIs, but, in some cases, the object is a literal string. The predicate provides information about the kind of relation that links the subject and object, acting as a verb in a natural language sentence. For example:

http://dbpedia.org/resource/Acanthemblemaria_atrata (subject)

<http://dbpedia.org/ontology/class> (predicate)

<http://dbpedia.org/resource/Actinopterygii> (object)

The subject is a URI that represents the fish species *Acanthemblemaria atrata* in the DBpedia dataset, the predicate corresponds to the biological taxonomic rank *class*; and the object refers to the element *Actinopterygii* in the same dataset. The previous triple translated in natural language means that the fish species *Acanthemblemaria atrata* is in the taxonomic class *Actinopterygii*. The use of URIs to identify the different elements that integrate the triple allows applications to process meaningfully this information by resolving the URIs and getting response data structured in RDF format describing those entities.

If the subject and object belong to different datasets, a typed link is created between the involved datasets. Therefore, two datasets are connected if one contains at least one triple where the object

is a URI belonging to the other dataset.

In the RDF data model, the statements can be represented as nodes and arcs in a graph: the subject and object are nodes and the predicate is an arc. For example:

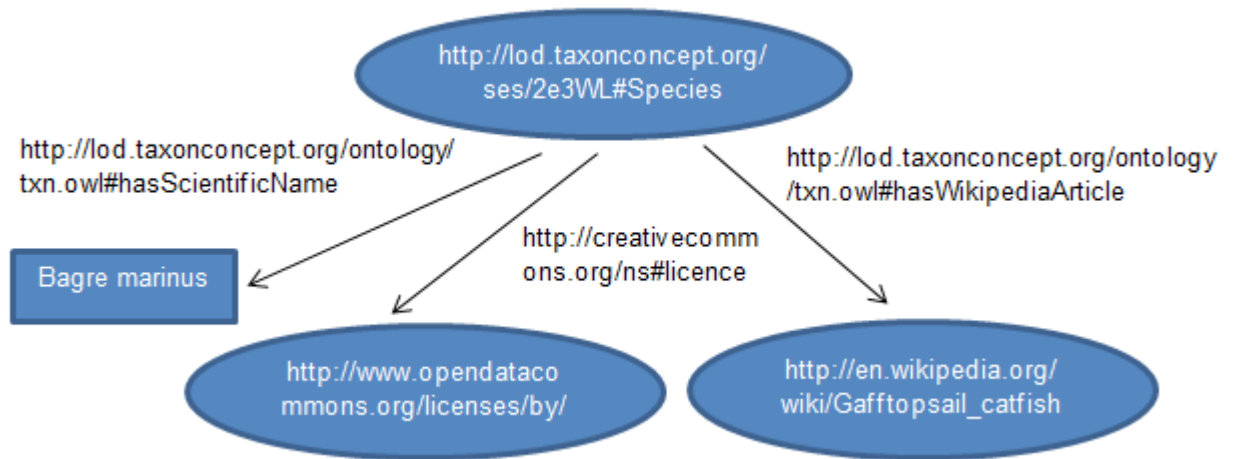


Figure 1: RDF graph example

In the example above, three statements are provided about the entity represented with the URI `http://lod.taxonconcept.org/ses/2e3WL#Species`, which could be translated into RDF triples. The URI nodes are represented as ellipses, and the literals as boxes.

An RDF document can be serialised in different formats (RDF is only a data model and not a file format). These formats are detailed below.

N-Triples

Each line of an N-Triple file contains an RDF triple followed by a dot. URIs are enclosed in angle brackets, and literals are double-quoted strings [9]. The advantage of this notation is its simplicity, but it is verbose. The RDF graph shown in the Figure 1 can be represented in N-Triple format as:

```
<http://lod.taxonconcept.org/ses/2e3WL#Species>  
<http://lod.taxonconcept.org/ontology/txn.owl#hasScientificName>  
"Bagre marinus".
```

```
<http://lod.taxonconcept.org/ses/2e3WL#Species> <http://creativecommons.org/ns#licence>  
<http://www.opendatacommons.org/licenses/by/>.
```

```
<http://lod.taxonconcept.org/ses/2e3WL#Species>  
<http://lod.taxonconcept.org/ontology/txn.owl#hasWikipediaArticle>  
<http://en.wikipedia.org/wiki/Gafftopsail_catfish>.
```


N3

N3 tries to solve the problem of the redundancy of data in N-Triples, which may not be a concern with small amounts of data, but can represent some problems while transmitting and processing large amounts of data [9]. Based in the fact that, commonly, a node is the subject of many triples, the redundancy of data is decreased by representing repeated nodes with a symbol (;). Additionally, this notation allows the use of namespaces and prefixes. The RDF graph example is represented in N3 in this way:

```
@prefix txn: <http://lod.taxonconcept.org/ontology/txn.owl#>.
```

```
@prefix cc: <http://creativecommons.org/ns#>.
```

```
<http://lod.taxonconcept.org/ses/2e3WL#Species>
```

```
    txn:hasScientificName "Tony Benn";
```

```
    cc:licence <http://www.opendatacommons.org/licenses/by/>;
```

```
    txn:hasWikipediaArticle <http://en.wikipedia.org/wiki/Gafftopsail_catfish>.
```

RDF/XML

Possibly the most popular format for RDF documents, the RDF graph is encoded in an XML document. Thus, objects, predicates and subjects are represented in XML terms: element names and contents, attribute names and values [10]. The example is serialised in RDF/XML format in the following way:

```
<?xml version="1.0"?>
```

```
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:txn="http://lod.taxonconcept.org/ontology/txn.owl#"
    xmlns:cc="http://creativecommons.org/ns#">
```

```
    <rdf:Description rdf:about="http://lod.taxonconcept.org/ses/2e3WL#Species">
```

```
      <txn:hasScientificName>Bagre marinus</txn:hasScientificName>
```

```
      <cc:license rdf:resource="http://www.opendatacommons.org/licenses/by/">
```

```
      <txn:hasWikipediaArticle rdf:resource="
```

```
        "http://en.wikipedia.org/wiki/Gafftopsail_catfish"/>
```

```
    </rdf:Description>
```

```
  </rdf:RDF>
```

In summary, RDF and URIs constitute an appropriate and standardised mechanism to represent data that has to be processed by computer programs in a distributed environment. This is the case with the WWW, as they provide a means to describe, identify, join and make assertions about entities in the Web.

The definition and organisation of the terms used for the construction of Linked Data assertions in the form of RDF triples is performed with the use of a schema language. Different schema languages

of this type are available. However, they have different intended purposes and the selection of one depends on the particular needs and characteristics of each scenario, as they provide different levels of expressivity, flexibility and logical support. The most relevant languages are described in the next sections.

2.2.2. RDF Schema

RDF defines only a data model, with no capabilities for the definition of vocabularies. RDF Schema is an extension to RDF to provide it with the capability to describe related resources and their relationships, with the use of a class and property system. This is similar to the system used in object-oriented programming languages [11].

The resources are grouped in classes and are instances of a class. RDF Schema can also describe the domain and range of a property. The main classes of this language are: *rdfs:Class* and *rdfs:Property*. For instance, it can be defined as a class *Book* with a property *has:Title*, and each specific book would be an instance of that class. Additionally, RDF Schema allows the specification of relationships between classes and between properties [2]. For example, it can be stated that a class is a subclass of another class or that a property is a subproperty of another property.

RDF Schema is appropriate to describe lightweight ontologies where subsumption relationships between terms are required [2], allowing the inference of implicit knowledge by appropriate tools.

2.2.3. OWL

The Web Ontology Language (OWL) is a knowledge representation language for the definition of ontologies, which are artefacts used to represent and organise knowledge in a domain by defining the relevant concepts and their relationships [12]. This language was designed to be used in situations where the knowledge captured in ontologies must be processed by applications. Those applications usually perform reasoning tasks based on that knowledge.

OWL is used to describe formally the meaning of a set of terms in a vocabulary and their relations, with the use of three main components: classes, properties and individuals.

Three different sublanguages with different levels of expressiveness are offered [13]:

- OWL Lite. Is suitable for providing a hierarchical classification using the basic features of the

language.

- OWL DL. Offers more expressiveness and all conclusions can be computed in a finite time.
- OWL Full. Provides maximum expressiveness but it is not guaranteed that all conclusions can be computed in a finite time.

OWL is a more expressive language than RDF, RDF-S and SKOS; offering more features to express meaning and semantics in machine-readable format [14]; providing reasoning support to validate the consistency of the knowledge captured in an ontology, and allowing the generation of inferences for the discovery of hidden knowledge.

Thus, OWL provides an adequate and powerful mechanism to structure hierarchically and define formally a group of terms from a domain of interest. Those terms can be used in Linked Data assertions in RDF files.

2.2.4. SKOS

The Simple Knowledge Organisation Scheme (SKOS) is a data model intended to capture structured vocabularies such as taxonomies, thesauri and classification schemes in machine-readable format [15]. This W3C recommendation is not a formal representation language, and is designed for the presentation of concepts organised in structures, where the organisation does not imply formal axioms or facts, and serves only as a convenient way to arrange those concepts.

As Bechhofer & Miles [15] highlight, some knowledge organisation systems (structured vocabularies) are not intended to provide a logical representation of the domain of interest, such as thesaurus, where logical entailments cannot be derived from the relationships between related concepts. Therefore, SKOS is a modelling framework more convenient in scenarios where a semi-formal and simple mechanism to define and structure a vocabulary is required, with no logical implications or interpretation. This is the case for a large number of relationships that join different entities in the Linked Data cloud.

The SKOS data model defines a knowledge organisation system as a set of concepts identified by URIs. These concepts can be labelled, grouped, documented with notes, and linked to other concepts with the use of semantic relation properties. SKOS data is represented as RDF triples.

Therefore, although SKOS is not as expressive as OWL, it offers other advantages that are valuable

in the Web of Data, such as the potential to define and organise a set of concepts, and establish associations among them in a semi-formal way where a logical perspective of the domain of interest is not required.

2.2.5. SPARQL

An important component of the Linked Data infrastructure and in the development of this work is SPARQL (Simple Protocol and RDF Query Language), a query language for RDF that offers a powerful and standardised method of extracting data from Linked Data resources.

Most Linked Data providers offer a SPARQL endpoint to perform queries against their datasets and extract data according to specific requirements. This is one of the fundamental benefits provided by the Semantic Web; the possibility of extracting valuable information from a site according to particular needs.

In a large number of cases, the information provided in a standard website is retrieved from a relational database. However, websites do not offer direct access to their databases to allow the extraction of data via SQL queries. Furthermore, even if they provide access to their databases, the schema of the data (diagram E-R) must be known in order to construct an effective SQL query.

In the Web of Data, the data is always structured using the same schema, the triple structure. As an example of the benefits of such query language, let us suppose that the list of fish species belonging to the class *Actinopterygii* and order *Perciformes* is required. If Wikipedia, or a similar resource, is used to obtain this information from the standard Web, there is no a direct and simple procedure to get such information unless a web page containing that specific information is published. Some websites may provide interfaces to filter data, but this is not a common practice. With the use of Linked Data technologies, this information can be extracted easily by selecting an appropriate dataset and constructing an adequate query. Clearly, a disadvantage of this approach is that the average user will not be able to construct SPARQL queries.

Generally, a SPARQL query is constructed with a group of triple patterns similar to RDF triples, but the subject, object or predicate can be variable [16]. Four types of queries are supported by SPARQL: SELECT, DESCRIBE, ASK and CONSTRUCT. The SELECT query is the type that has been most useful for the development of this project. The basic syntax of a SPARQL SELECT query [17] is:

```
SELECT some_variable_list  
[ FROM <some_RDF_source_URI> ]  
WHERE  
{  
    some_n3_triple_pattern .  
    another n3_triple_pattern .  
}
```

A select query is used to retrieve the solutions that satisfy a set of triple patterns. For example, the following query would retrieve all the predicates and objects of the triples that have the URI <http://lod.taxonconcept.org/ses/2e3WL#Species> as the subject:

```
SELECT ?p ?o  
WHERE  
{  
    <http://lod.taxonconcept.org/ses/2e3WL#Species> ?p ?o  
}
```

Some keywords are available to produce more accurate solutions: **DISTINCT** (removes duplicate solutions); **ORDER BY** (establishes the order of a set of solutions); **OPTIONAL** (specifies optional parts of the graph pattern); and **UNION** (combines the solutions of different graph patterns).

The queries that have been created for the development of this project are very simple and the use of advanced features of this language was not necessary.

It is worth mentioning that SPARQL queries are executed against a SPARQL endpoint, which is a service that data providers usually offer to their users (applications and humans) to enable them to find and retrieve RDF data. The results of the queries can be retrieved commonly in different formats (RDF/XML, XML, N3, HTML, etc.).

The SPARQL endpoints are one of the components of the triplestores, which are repositories in which RDF data is stored. A number of triplestore implementations are available in the market, and the specific services offered may vary. The triplestore used in this project was OpenRDF Sesame, which is one of the most popular implementations. It supports the SPARQL query language and can be populated with RDF data in all the main RDF formats, including RDF/XML, Turtle and N-Triples [18].

In summary, Linked Data applications can use SPARQL to extract information from RDF repositories and present useful and relevant results to the user. Consequently, the possibilities emerging from the use of a standardised RDF query language in terms of consumption of the information are extended, thus enabling the generation of *ad hoc* outputs by applications and advanced users.

2.3. Integration of Datasets

The main objective of this project is the integration of different fish species datasets; and, as discussed previously, the connection of two datasets is performed with the generation of RDF links containing URIs corresponding to elements from different resources. Therefore, it is worth noting some reasons and approaches to construct those links.

The first point for discussion is the value of creating links to external resources. In the standard Web, hypertext links are fundamental artefacts that provide the simplest and most natural means for exploring and navigating the Web. In the same way as the standard Web, Linked Data links offer efficient methods for the discovery of further data by applications and by humans using Linked Data browsers.

Moreover, the Web of Data shares a key property with the Web of Documents: publishers can link their data to any external resource they consider, and, conversely, publishers cannot control the sources with which their data is linked [3]. This allows for a free integration of resources in the Web, which encourages the creation of a single global space where data is interconnected; the Linked Data cloud.

From the user's perspective, a collection of links to external resources offers the possibility of complementing or extending the required information just a click away, whether he or she is navigating the standard or Semantic Web. It may be argued that, with the use of search engines, it is possible to find the desired data or information directly, without the need of links connecting different resources. However, without the existence of these links, every time that it was required to consult a related resource, it would need to access the search engine once again and start a new search by typing the key words. Therefore, search engines are powerful tools to find a starting point for crawling the Web, but the links serve as a convenient mechanism to navigate between related resources.

Furthermore, as previously mentioned, one of the four basic principles of Linked Data outlined by

Tim Bernes-Lee in 2006 is the creation of links to other URIs, thereby enabling the discovery of more data: “The Semantic Web isn’t just about putting data on the web. It is about making links, so that a person or machine can explore the web of data.” [1]

It might be said that the creation of links to external datasets adds little value to the data. If two web documents with the same information are compared, one containing links to related external resources and the other with no links, just plain text, it is clear that the document with links is more valuable as we can easily extend the information and create a broader picture of the topic with little effort. This comparison can be applied to Linked Data RDF documents.

The generation of RDF links is performed usually with the use of automated or semi-automated procedures, as data sources commonly contain large numbers of entities [4]. The first task usually consists of finding the external datasets to be linked to the pivot or base dataset. Subsequently, selection of the predicates that will be used to construct the RDF triples is required. The final step involves the creation of the links, and the necessary identification of the elements to be linked. A common practice is to link URIs that refer to the same concept.

The main challenge in the generation of links is the identification of the entities from the different data sources that correspond to the same concept or object of the real world. There are no standardised procedures, tools or guidance to perform this identification. A manual identification of entities could solve the problem, but this is impractical and time consuming when the amount of data is large.

A number of papers have been published discussing the difficulties that arise when linking datasets and the different approaches to solve them.

Liu et al. [19] analyse the problem of finding the two entities in the datasets that describe the same object, calling it the “fusion problem”. They suggest that it can be partially solved with the use of techniques developed in related areas, such as sequence alignment, a string comparison algorithm used to find the similarity of DNA sequences. Additionally, this paper lists the following frequent problems to perform this detection, and proposes some solutions:

- Two entities can be described in different languages.
- Two semantically-equivalent entities can have different literal contents, where a string comparison algorithm is not useful, which is the case of synonyms. The authors suggest the use of thesaurus to solve this problem.

- Hardly two entities can be compared based only on their URIs, requiring to deference the URIs to extract information that can be used in the comparison.

Hassanzadeh & Consens [20] published a paper presenting the challenges and solutions in the development of the Linked Movie Database project for the connection of Linked Data film resources. It identifies that the access to the data in the target data source can be limited, and discusses that the literal matching of strings is not an adequate approach in that domain, as film titles such as “A Thousand and One Nights” and “1001 Nights” would not match. On the contrary, approximate string matching algorithms could generate false matches of film titles, and it is provided as an example the case of the strings “Face to Face” and “Face to Fate”, which clearly correspond to different films.

A number of functions for comparing the similarity of strings are presented in that paper, and their accuracy is evaluated for that specific case study. It concludes that a combination of an appropriate string similarity algorithm and techniques to compare additional co-occurrence information can reduce the number of false matches. However, as this approach does not guarantee complete accuracy, they propose the inclusion by the publisher of metadata about the links and how they were generated. This allows the users to evaluate the quality of the links, and provide feedback to the publisher.

The paper published by Behkamal et al. [21] discusses the encountered problems of publishing and linking academic data, and they highlight that interlinking datasets represents one of the main challenges in Linked Data, as simple literal string comparisons produce poor results. As an example, it is described the case of linking geographical data with Geonames, where a query to retrieve data about the city of Vienna returns 20 results, as there are many cities with that name in the world.

Another important challenge described in that paper is the selection of adequate ontologies and predicates. It indicates that, generally, ontologies are selected based on their popularity, as some have become standards in specific domains, such as FOAF (Friend of a Friend) for personal information or Dublin Core for data about publications. Nevertheless, it concludes that this approach is not useful in domains that lack a well-known ontology, and the identification and selection of candidates cannot be performed automatically.

A semi-automatic approach is used in the work described above, where no *de facto* standard ontology exists in the domain of interest. This consists of searching for ontologies containing the required concepts with the use of semantic search engines, and estimating their popularity by executing SPARQL queries in the Linked Open Data cloud (LOD) endpoint to know the number of

times that the predicates of the prospective ontologies are used in the cloud. Thus, the ontology with more occurrences is selected.

The selection of appropriate predicates is another important step in the construction of links between datasets. A common practice is to link different datasets by constructing links between URIs that refer to the same real world object, called URI aliases. Bizer et al. [4] highlight that URI aliases are very common in the Semantic Web because data providers could hardly agree the same URI to identify the same real world element.

The most common predicate used to link URI aliases is *owl:sameAs*, which indicates that “two URI references actually refer to the same thing: the individuals have the same identity” [14]. A considerable amount of literature specifies that the use of the *owl:sameAs* predicate is the most appropriate and natural choice for the integration of URI aliases. However, a growing number of authors question the use of this predicate for this application, as it could lead to logical inconsistencies.

As an example of the overuse of this predicate, Jain et al. [22] provide the case of the instances of Barcelona in DBpedia and Geonames, which are linked with the use of *owl:sameAs*. The population of the city is different in those datasets, which represents a logical inconsistency that could be detected by a reasoner. Alternative predicates are offered in a large number of available vocabularies. In particular, SKOS provides a wide variety of predicates to make connections between entities with different levels of integration, avoiding that kind of inconsistencies.

Another important decision that must be taken in the generation of links is the selection of the target datasets. In this regard, several factors must be considered: coverage, popularity, quality of the data and ease of access to the data, among others.

The following considerations are suggested by [2] to be considered in this selection:

- The value of the data in the target dataset
- The level of maintenance and the stability of the ownership of the target dataset
- The probability of change in the URIs
- The number of links to other datasets that the target dataset contains, which indicates the degree of integration of the resource in the Linked Data cloud.

In conclusion, a number of challenges must be addressed for the integration of datasets; crucially, the selection and implementation of the mechanism required for the identification of entities in different datasets that describe the same real world concept, the discovery and selection of appropriate ontologies and predicates, and the finding and selection of adequate target datasets. The different approaches to those challenges depend on the particular characteristics of the domain of interest and datasets. However, the reasons for the creation of links are enough to justify the effort that represents those challenges.

2.4. The Web of Data

Millions of RDF assertions published by a large number of data providers in the Web integrate the Web of Data.

A functional Web of Data is dependent on the availability of large amounts of machine-readable data [23] , but in 2007, the amount of information published as Linked Data was limited. In that year the W3C Linked Open Data (LOD) project [24] was created to bootstrap the Web of Data by publishing a considerable number of existing datasets as RDF triples and generating links among them, according to the principles outlined by Berners-Lee. Since 2007, the Web of Data has been growing steadily, integrating data from a broad range of topics.

The LOD cloud [25], generated in November 2010 and presented in the Appendix A, shows the diversity of organisations that have contributed to its expansion, with the publication of data covering a wide variety of branches of knowledge: geography, medicine, music, chemistry, biology, etc.

Some of the most popular data sources in the Web of Data that are analysed in this project are detailed below.

Freebase

It is a cross-domain popular dataset containing data on approximately 20 million entities associated with people, places, books, films and other topics, including a collection of animals [26].

Geospecies

Contains data on 19,230 species, and is linked to Freebase, DBpedia and other popular RDF data sources such as Geonames and Uniprot, according to information in its website [27]. One of the main challenges of the project is the creation of unique identifiers that are not altered by changes in

the taxonomic classification of species.

BBC Wildlife Finder

This broadcasting corporation traditionally has published large amounts of content online in the standard Web of Documents. Since 2007, the BBC started to publish content in RDF format about programmes, music and species, among other topics. BBC Wildlife Finder combines zoological data from different resources, providing a URI identifier for each species, and is linked to clips from the BBC nature archive [28].

DBpedia

DBpedia is undoubtedly the most popular cross-domain dataset which interconnects a significant number of Linked Data resources in the Web, acting as the centre of the LOD cloud. The aim of DBpedia is to provide structured information from Wikipedia covering a wide range of domains, and currently contains data for more than 3.5 million entities, including 169,000 species [29] [30].

The characteristics of these datasets are discussed in more detail in a later chapter of this paper.

2.5. The FishDelish Project

FishBase is possibly the most comprehensive database of fish-related information available on the Web. The website of this project [31] states that it has information about 32,000 species, with 291,200 common names, 50,400 pictures and 45,800 references, and it is curated with the aid of 1,850 collaborators (May 2011). Moreover, it claims to contain almost all known fish species. This project was developed by the World Fish Center (a non-profit international organisation that researches fisheries) and the FAO (Food and Agriculture Organisation of the United Nations). Currently, it is managed by nine research institutions, including renowned international museums and universities.

FishBase is structured as a relational database which is exploited via a web interface, allowing the search of information with the use of keywords and filtering it by taxonomic family, country, ecosystem and other criteria.

FishDelish is a project developed from June 2010 to March 2011, led by the University of Manchester, with the purpose of exposing a large amount of the FishBase database as Linked Data [32]. The data from the most popular categories of FishBase, including taxonomy, pictures, distribution and references, was translated into RDF format, according to the FishDelish Project

website [33].

“A key task of the FishDelish project is to produce applications which exploit the LD version of the FishBase data in interesting ways” [33]. Some examples of these applications are described in the site of the project, one of the most relevant being the case of other fish species’ databases (aquariums, museums, shops, personal web pages etc.). With FishDelish, the users have more tools to exploit the vast amount of information in FishBase, so they can extend the content of their collections or use FishBase as the main source for their databases [33].

The architecture of FishDelish comprises the following main components:

- A 4store triple store, which stores RDF data and provides a SPARQL endpoint to allow the retrieving of data.
- The species browser; a PHP web application which uses some special libraries to enable the interaction with RDF content, running on an Apache web server, providing the standard web interface to users. The PHP application extracts data via SPARQL, queries and transforms the results into HTML code with the use of XSLT transformations.
- `rdf_generator`, a Ruby on Rails application that allows the dereferencing of URIs. This application generates the RDF content of the requested URI by executing a CONSTRUCT SPARQL query and returning the result as a response to the request.
- `ldoc`, another Rails application, supports the generation and publication of documentation.

As part of the development of FishDelish, other interesting tools have been developed. For example, FishdelishSPARQL Explorer is an application that can be used to learn SPARQL by browsing, studying, executing, modifying and analysing the results of a collection of queries. FishOMatic is another, which is described as an application that “allows users to easily utilise FishBase data on their own pages to produce an observation log” [33].

As of May 2011, FishDelish contains information about 31,927 fish species. This dataset has been used as the central hub for the creation of links, as this dissertation project seeks to construct links connecting fish species entities belonging to the FishDelish dataset with other public datasets available in the WWW.

2.6. The FishLink Project

The main purpose of this project is the development of tools to enable freshwater biologists the publication of information as Linked Data [34]. Some databases in spreadsheet files, text files and

other formats were converted into RDF format in order to generate a central repository of information. The creation of links connecting related entities is another important aim of the project, to allow the analysis of information extracted from different resources in a single dataset, as the research conducted by freshwater biologists requires the analysis of a variety of information from different resources [34]. Location of bodies of water, climate records, observational records and species data are some of the information integrated in this project.

Some of the challenges that had to be addressed during the development of FishLink included the identification of overlapping data while merging separate datasets; the extraction and normalisation of data from heterogeneous resources; the record of versioning and provenance of data; and the identification of linking points.

One of the initial objectives of this dissertation was to use the FishLink and FishDelish datasets as the base datasets for the generation of links to other data sources. Unfortunately, the species data in FishLink was not totally available during the development of this work. Consequently, FishDelish is the only base dataset used in this project for the creation of links to other public datasets available on the Web.

2.7. Other Resources of Species on the Web

Unsurprisingly, virtually all of the most popular and comprehensive resources of species information available in the Web are not exposed as Linked Data. Most consist of relational databases exhibited through standard web front-ends. For our purposes, two kinds of repositories can be identified: databases containing data of species from all the biological kingdoms and databases specialising in fish species. The following databases are some of the most important resources of the first category:

ITIS (Integrated Taxonomic System)

This provides taxonomic information of plants, animals, fungi and microbes of North America and the rest of the world. It was created in the mid-1990s by several US federal agencies to collect and analyse taxonomic information [35]. As of March 2011, it contained 527,310 scientific names.

The web interface allows the retrieval of data for specific taxonomic groups in text files, and the whole database can be downloaded in two formats; a Microsoft SQL Server 2000 Database file and an Informix Database file. Additionally, a web service is offered to explore the database and retrieve data.

Species 2000

The main objective of this project is the creation of a list including all the species of plants, animals, fungi and microbes that exist, by integrating in a central database a set of specialised databases containing data about the major groups of organisms. Currently (April 2011) Species 2000 contains data from 52 different databases, bringing together 60% of the total known species [36]. This project began in the early 1990s as a joint programme between several research organisations of biological sciences.

Catalogue of Life

Species 2000, in conjunction with ITIS, developed Catalogue of Life with the purpose of creating the most comprehensive information system of species. In 2001, these two organisations decided to work together to create a centralised repository covering the 1.75 million known species of organisms on Earth. Catalogue of Life is the result and it has 1,368,009 species in its last edition (July 2011) and it compiles 100 taxonomic databases about different types of organisms. The information in this database is shared as a free DVD or can be accessed through the website [37].

Encyclopedia of Life

The main purpose of Encyclopedia of Life is also the generation of a database containing all the species of organisms that are currently known, enabling a better comprehension of the life on Earth. It is another collaborative project developed by a number of research institutions based primarily in the US. The architecture of this project is supported on a PHP, MySQL, Ruby on Rails and Linux platform. It integrates information from a number of resources, especially from the Catalogue of Life, but the fish species data is taken from the FishBase database [38]. It contains “more than 1,900,000 pages on January 7th, 2010, including living and extinct species, families, orders and so on” [39].

GBIF (Global Biodiversity Information Facility)

Just like the Encyclopedia of Life, it integrates information from a large number of data providers (323), its main source of data being the Catalogue of Life, and contains 293,485,946 records (July 2011), including species, locations, observational records and other related information. “One of GBIF’s main purposes is to establish a global decentralised network of interoperable databases that contain primary biodiversity data” [40]. It is an international organisation, initiated and funded by government institutions from all over the world, and its main objective is to make biodiversity data freely available to the scientific community and to any other kind of users. [40]

As it can be noticed, there is a close relationship among the most important species repositories offered in the Web, where some databases are fed with data from other repositories. Species 2000

and ITIS provide the vast majority of data used by Catalogue of Life, which, in turn, is the main data provider of the GBIF and Encyclopedia of Life databases. This is illustrated in the figure below.

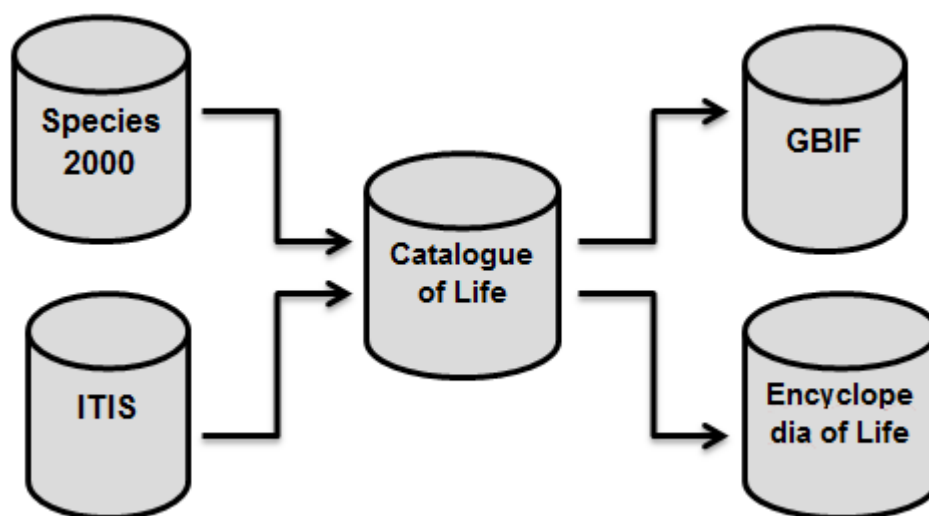


Figure 2: Relationship among the most popular species repositories

With regard to databases specialising in fish species, a large number of collections can be found on the Web from museums, universities and other institutions, but, generally, their coverage is small. The most visible and possibly most comprehensive resource on the Web is FishBase. Nevertheless, it is worth noting two relevant datasets that contain a large collection of fish:

FishWise

This is a relational database that claims to contain more than 98,400 species and more than 34,000 pictures (July 2011). The number of species is doubtful given the fact that different authors specify that the number of known fish species is no more than 40,000 [41] [42]. One possible explanation is that this also includes other water-dwelling species. The source databases of this site are FishBase, Encyclopedia of Life, the Catalogue of Fishes of the California Academy of Science, and other resources. The site was created to gather a collection of photographs taken by divers and intended to be used by anybody who is interested in fish and their habitats [43]. This is a very user-friendly website that offers different criteria to filter and find data by taxonomic classification, common and scientific names, names of authors, countries, photographers, etc.

Catalogue of Fishes of the California Academy of Sciences

According to information in this website, the Catalogue of Fishes has been the main resource of taxonomic information for FishBase since its creation. This collection has been maintained for 25 years, and the authors of this work consider this database to have the most current and accurate number of fish species, because it tracks all new species and revisions on a daily basis. The last

version available of this database (14th of July 2011) contains 56,712 names of species, 31,938 of which correspond to valid species (species recognised in the literature) [44].

2.8. Biological Background Information

This project is highly related to fish, and even though analysing fish species from the biological perspective is not an objective of this dissertation, some important biological information must be offered to readers who are unfamiliar with Taxonomic concepts.

2.8.1. Taxonomic Classification of Fish

Sumich and Morrissey [45] comment that the taxonomic system of classification provides a methodology to organise the vast number of species of organisms and their relationships. They indicate that there are between 10 and 30 million of species currently on our planet, but only around 1.9 million of them have been identified and studied. The basic element in the taxonomic classification is the species, which is “a group of closely related individuals that are similar in appearance and that can and normally do interbreed and produce fertile offspring” [45]. The taxonomic classification system consists of a hierarchical structure of categories (taxonomic ranks):

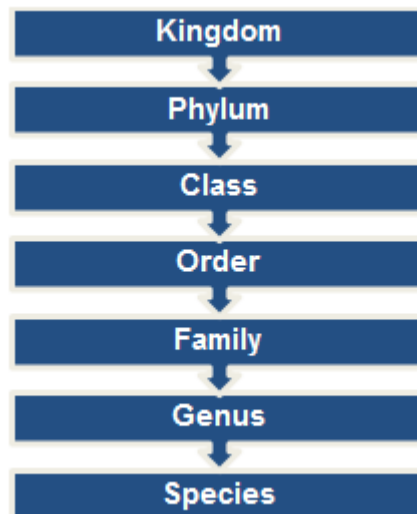


Figure 3: The taxonomic classification system

Each category subsumes under it a number of more specific taxonomic ranks. Some literature and other resources of information use intermediate ranks, such as Subfamily or Subphylum.

As an example of the taxonomic system, below is the classification of the fish species *Astronotus ocellatus* (Oscar fish):

Kingdom: *Animalia*
Phylum: *Chordata*
Class: *Actinopterygii*
Order: *Perciformes*
Family: *Cichlidae*
Genus: *Astronotus*
Species: *Ocellatus*

Hickman et al. [42] point out that, presently, the modern concept of “fish” is used for convenience, but it does not represent a taxonomic group of species. They also explain that the word “fish” refers to a mixed collection of water-living animals in a common and older sense. Today, it is known that even when some animal species are named *x-fish*, such as the jellyfish, the starfish and the shellfish, they are not actual fish. But some centuries ago the biology did not make that distinction, and other sea creatures were classified as fish, including seals, whales and amphibians, according to [42].

Today, however, the concept of fish refers to “an aquatic vertebrate with gills, appendages, if present, in the form of fins, and usually a skin with scales of dermal origin” [42]. Another way to define the term fish is “all vertebrates that are not tetrapods” [42], which has an evolutionary connotation.

Hickman et al. [42] also report that fishes are the most ancient and diverse vertebrates, with around 28,000 living fish species, out of 55,000 living vertebrate species. The following classification of major fish taxa is presented in their study, where five of the nine living vertebrate classes correspond to classes of fishes. However, the authors point out that other schemes of classification have been proposed.

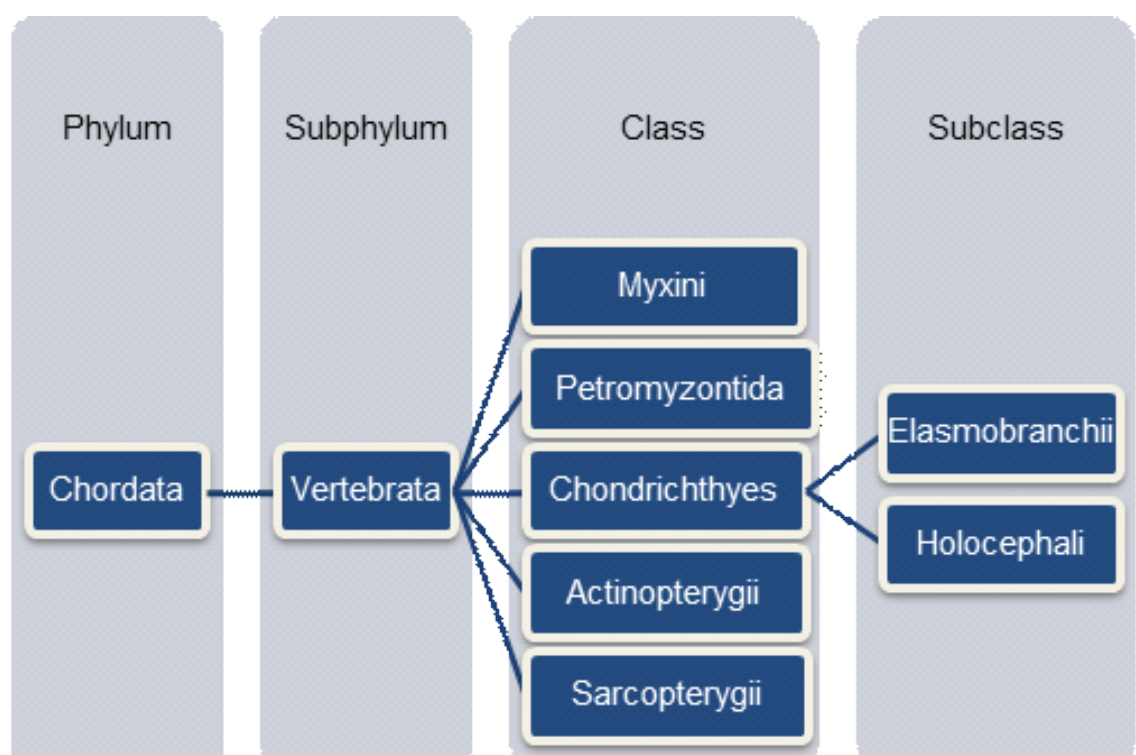


Figure 4: Classification of fish species

R. L. Kotpal [41] also agrees that there is no universally accepted fish classification, which is a consequence of the huge quantity of species (he claims that 40,000 species are known) and the diversity of their characteristics in terms of shape, size, behaviour, etc. In his text he adopts the classification of fish suggested by Parker and Haswell in 1960 with some small variations, which consists of three classes that belong to a superclass called *Pisces*. According to him, this classification has been followed, with some minor changes, by all the eminent authors in the field of zoology.

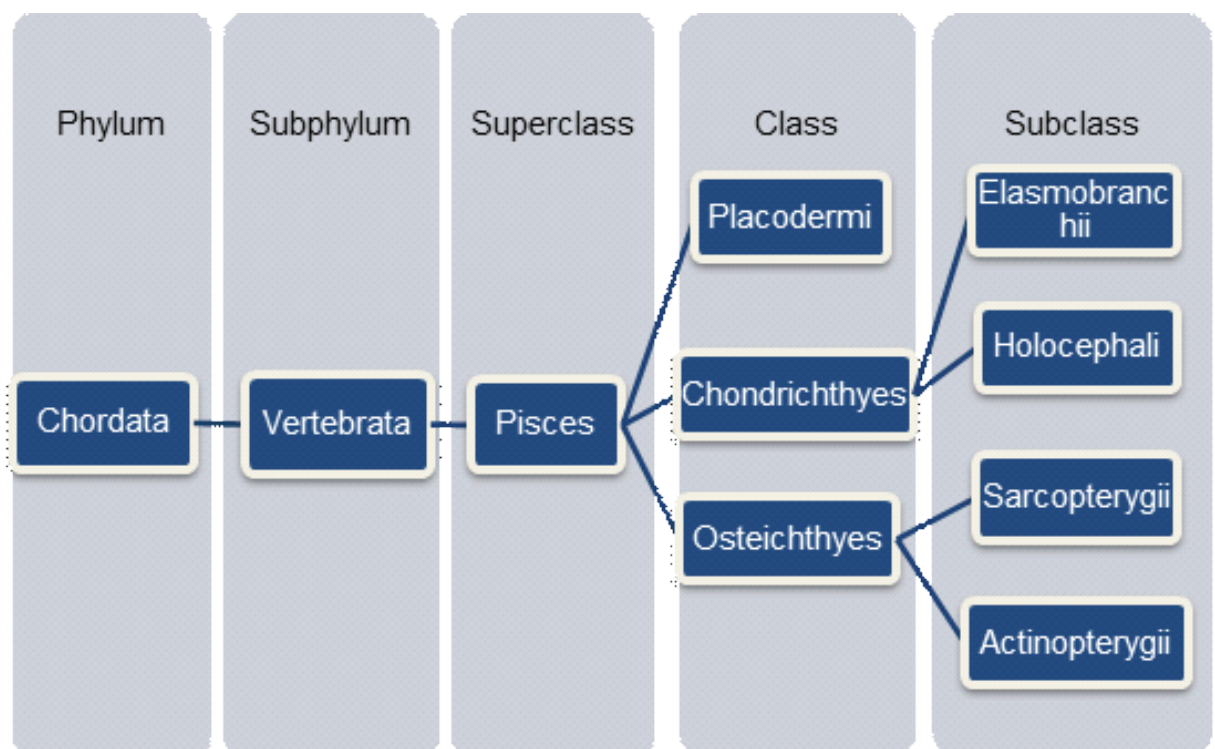


Figure 5: Classification of fish species 2

As it can be noticed, the *Sarcopterygii* and *Actinopterygii* are subclasses in this classification, instead of classes, as suggested by Hickman et al. Additionally, the *Myxini* (hagfish) and *Petromyzontida* (lampreys) classes are not considered as part of the *Pisces* superclass. The lampreys and hagfishes are classified in the class *Cyclostomata* by R. L. Kotpal.

This disagreement in the rank of certain taxonomic clades (taxonomic groups of species) is presented frequently in the literature and, consequently, in the information available on the Web. While some resources may consider the taxonomic clade *Actinopterygii* as a class, other resources may define it as a subclass. This problem can also be observed with other important fish taxonomic groups, such as *Elasmobranchii*, while comparing different resources of information.

It is important to note that there are some groups of species that are more relevant for our purposes, as the datasets used in this project, generally contain a much larger number of entities from some clades than from others. As an example, 8,174 out of the 9,145 fish species found in DBpedia correspond to species belonging to the class *Actinopterygii*, while Geospecies contain 25 entities of the same class out of 28 fish species.

2.8.2. Scientific Names

The scientific name is the standard means of uniquely identifying biological species (and taxonomic clades) in the science, as common names may refer to more than one species, and even when a common name refers to one species, it can vary depending on the country or language, according to Armstrong [46]. Bailey and Burgess [47] comment that a common name might be used in different languages referring to different fish species, and provide as an example the case of the name “zebra cichlid”, which refers to *Metriaclichia zebra* (an African species) for English speakers, while in German denotes *Archocentrus nigrofasciatus* (a Central American species).

According to Bailey and Burgess, the two key properties of a scientific name are uniqueness and universality. The first property indicates that a scientific name can only be assigned to one taxa, and the universality indicates that it is valid in all countries, regardless of language. A species scientific name is written in Latin and is composed of two parts: the genus (which identifies groups of closely related species) and the species name. This naming scheme is called binominal nomenclature.

Even when scientific names are unique, they can change, which may result in more than one scientific name for a particular species. The most recent name is the valid one, but outdated scientific literature may contain the old name, according to Dewey [48]. Outdated scientific names are commonly referred to as synonyms.

In conclusion, the knowledge of the different biological concepts and facts explained in this section regarding the taxonomic classification and nomenclature of fish species is useful in understanding the content and structure of the datasets analysed in this dissertation, as well as some elements of design in the implementation of the project.

In particular, it is necessary to be aware of the differences in the taxonomic classification of species presented from one resource to another. Those differences are especially important in the generation of SPARQL queries that are constructed to filter the fish species in datasets containing data about

other types of organisms, as a wrong query may represent the retrieving of incomplete data.

Finally, the understanding of the common and scientific names allows the selection of appropriate mechanisms to identify URI aliases for the construction of accurate links.

2.9. URIs in the Biodiversity Domain

In the biodiversity domain, the most common way to find, retrieve and organise information is based on the use of scientific names. The scientific names generally serve as the primary identifiers of species in the databases containing biodiversity data.

A common problem in this field is the integration of data from different resources. This problem has been solved by the use of the scientific names as identifiers to link records.

However, as Page [49] points out, the use of scientific names as identifiers has some disadvantages, because they are neither stable nor unique. As described in the previous section, the scientific names may change as a result of taxonomic revisions, and some species may have more than one name.

DeVries [50] explains that one of the main challenges in the field of biodiversity informatics is the creation of a system of identifiers for species separated from taxonomic concerns. According to him, the binominal system for naming species (genus and species) has two objectives. The first is to indicate the taxonomic classification of species by stating the genus that the species belongs to. The second objective is to assign a “universal stable identifier” to the species. However, DeVries argues that the two roles are incompatible: “How can a species have a stable identifier when a taxonomic revision will change the identifier itself?” [50].

Another problem identified by Page [49] is that the number of data providers of biodiversity information is large and the topics covered are diverse, ranging from taxonomic information to museum collections. Additionally, those databases contain data that was available at a certain time, and hardly that information is updated to reflect changes in the nomenclature of species and taxonomic groups.

Consequently, the use of URIs according to the principles of Linked Data provides an alternative solution to solve the problem of the identifiers of species. Thus, the solution comprises two elements:

- Assigning identifiers to the species that are not related with their nomenclature. For example, URIs containing a random set of alphanumeric characters.
- An agreement among the different data providers to use the same identifiers to allow the integration and sharing of data across different data sources.

Page also comments that two basic properties that must have a GUID (globally unique identifier) are persistence and availability, which are met by the URIs.

Most integrations of biodiversity databases that have been produced so far (i.e. GBIF) have been conducted with the use of binominal names to link records, according to [49]. Nevertheless, the use of adequate URIs as identifiers in biodiversity data sources can increase and simplify the integration of different resources of information. “What is needed in the digital age is a commitment to deploy and reuse globally unique,shared identifiers” [49].

3. IMPLEMENTATION

As discussed previously, different approaches to solve the challenges arising from the integration of datasets can be applied according to the particular characteristics of the domain of interests. However, a common denominator in the analysed literature is the general methodology followed in performing this integration, which may differ in the specific mechanisms selected to resolve some technical difficulties.

Therefore, the following procedure has been designed based on the general approach suggested by Heath and Bizer [2] for the creation of links with external data sources, and on the procedures conducted in other projects to link datasets [19] [20] [21]. However, at the same time, this takes into account the particular characteristics of our domain of interest for the selection of the specific techniques employed in the different stages during the development of the project.

The different stages of the implementation are described briefly below:

- Identification of prospective datasets. One of the most important stages in this project is the identification and subsequent selection of appropriate datasets available in the Linked Data cloud, as a satisfactory conclusion of the work depends greatly on the size and other characteristics of the data that our base dataset is linked to.
- Analysis and selection of datasets. Once a group of prospective datasets was identified, a procedure must have been applied to determine which satisfies our requirement and interests. A decision matrix was used to identify the strongest alternatives, requiring the evaluation of three different characteristics of the datasets.
- Selection of predicates. As the main purpose of this project is the integration of fish species' datasets, the links that are generated correspond to links connecting URI aliases of fish species between the base and target datasets. In this regard, a discussion has been initiated in the background section about the most appropriate vocabularies and predicates for this purpose, concluding that even when the most common predicate used in this scenario is *owl:sameAs*, predicates from other vocabularies may provide a more appropriate connection between the involved datasets. Thus, a procedure was applied to identify the most adequate predicates.
- Generation of links. Due to the large number of triples generated, an automatic approach was used for the creation of the links, based on a suitable record linkage technique.

- Development of the application. The final stage of this study consists of the development of an application to demonstrate the use of RDF links in applications that combine data from several resources.

Each stage is described in more detail in the corresponding sections of this chapter.

3.1. *Dataset Comparison Application*

A Java application was developed to perform different tasks in the development of this project, including the identification of URI aliases by comparing the scientific names of two input files. This application is also used later to generate the set of links. The reasons for the selection of this programmatic approach instead of the execution of federated SPARQL queries (queries to retrieve and combine data from different datasets) to perform the comparison are set out below:

- Not all datasets provide a SPARQL endpoint, and, in other cases, the number of results that can be retrieved is limited to a small number, which makes the extraction of data difficult.
- Some operations are required to normalise the strings being compared. While it is true that this string pre-processing might be performed with SPARQL, the consequence of this approach would be the generation of very complex queries.
- With a programming language such as Java, there is more flexibility to manipulate data than with a query language such as SPARQL; thereby allowing the generation of the required output files containing the RDF links with less effort from a technical perspective.

During the second stage of the project, which involves the selection of datasets, this application was used to count the number of overlapping species between the two input datasets. FishDelish was always one of the input datasets, and the second was any of the other data sources found during the first stage of the project (Identification of Datasets).

The inputs of this application are two files in any format that can be processed with Java, which must contain a list of Linked Data entities representing biological species. Those entities must have at least two pieces of data: the URI of the entity and the scientific name of the species in binomial nomenclature. In generating the RDF links, the authority of the species is also required.

The key element considered in the design of this application was the ability to compare datasets in different formats, as the input datasets might be presented in a wide range of formats, such as

RDF/XML, N3, or even CSV and spreadsheet.

Another important element considered in the design of this software was its scalability, as this application was used to compare several datasets with the base dataset (FishDelish). Consequently, this application must have been designed to require minimum modifications when a new dataset is compared with the base dataset.

Finally, another relevant principle in the design of this application is its modularity. It needs to encapsulate the logic required to parse each dataset and to perform other operations in order to keep a clean, understandable and flexible code. Different functionalities were used in different stages of the project, such as the generation of the RDF files containing the links or the comparison of the authority with more advanced techniques. Consequently, the ability to add or remove functionalities was eased by a modular design.

The following design is the product of the previous requirements. The main characteristic of this application is that for each new dataset that must be compared, a new Java class must be added, which encapsulates the logic to parse and extract the data of the input file and generates a Java list that can be manipulated in the rest of the application.

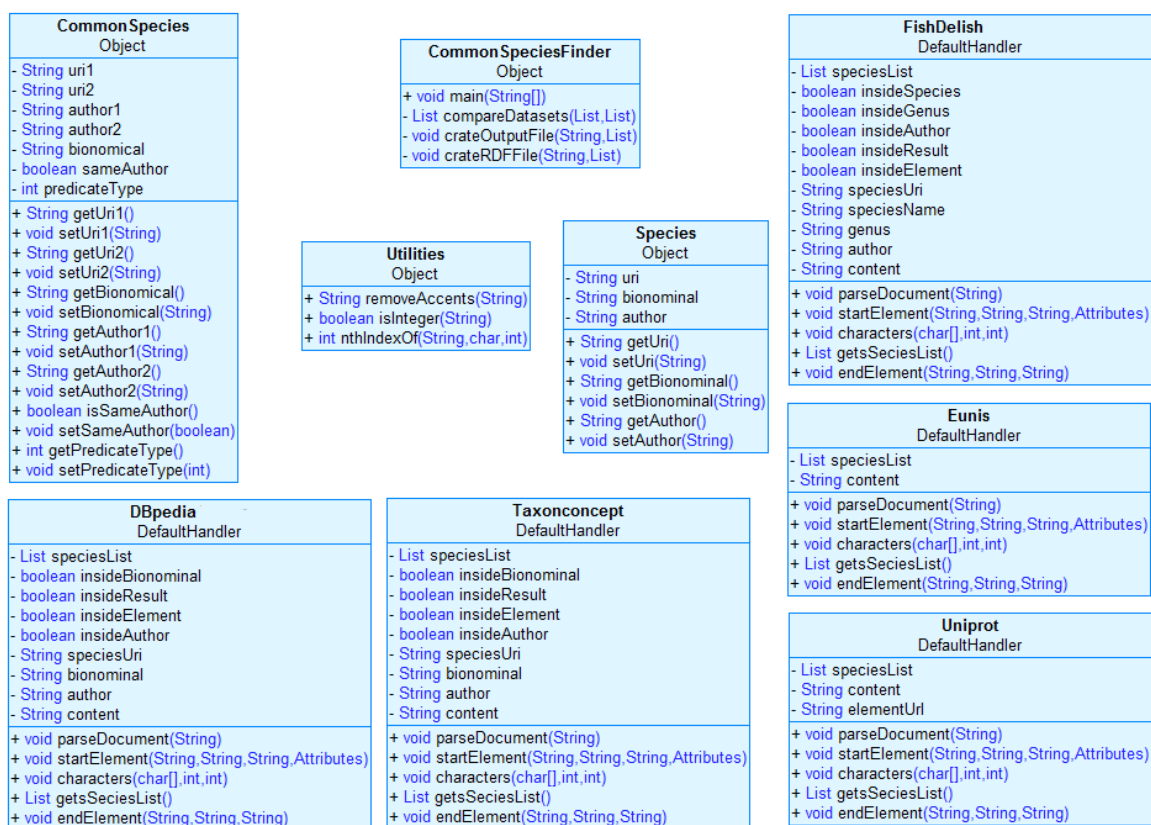


Figure 6: Dataset Comparison Application class diagram

The main class of the application is *CommonSpeciesFinder*, which contains the method *compareDatasets*. This method receives two objects of the class *java.util.List*. Each list contains a set of objects of the class *Species*, and compares the two lists to extract the elements that refer to the same species (URI aliases). The output of this comparison is another Java list of objects of the class *CommonSpecies*; a Java bean that stores information of the overlapping species.

The lists that serve as input of the method *compareDatasets* are the outputs of the methods *parseDocument* contained in the different classes created to encapsulate the logic to process the input files.

The first task performed by the application is the extraction of data from each input file in order to convert it to a Java list of objects of the class *Species*. This is a Java bean with three properties (URI, binominal and authority).

The procedure to count the number of overlapping species involves taking each entity from the first dataset and comparing its scientific name with the scientific name of each entity in the second dataset, until it is found. The comparison of scientific names is a literal comparison of strings, as all the binomial names have the same structure: two words separated by a white space or other character. However, a pre-processing of the names is applied before performing the comparisons to normalise the strings, which consists of trimming white spaces at the end and the beginning of the string, converting all the characters to lower-case and converting the character used to separate the two words integrating the binominal name into a white space, as other characters such as the underscore are used in some datasets. The procedure to compare the authority of two species is different, as a literal comparison of string is not useful to compare names of people and years, so other techniques that will be described later must be applied.

The output of the application is the number of species contained in each input file and a Java list of *CommonSpecies* objects. This is used to discover the number of overlapping species between the two datasets being compared during the stage of selection of datasets, and to generate the files with the RDF predicates in the link construction stage.

3.2. Identification of Prospective Datasets

As previously discussed, the main purpose of this project is the construction of RDF links connecting the fish species within the FishDelish dataset, with the species from other public resources available

in the Linked Data cloud. Thus, the first stage of the project involves the identification of datasets containing data of fish species in RDF format. The amount and quality of fish species' information provided by the datasets is not important at this point. Any data source containing any kind of fish species data is taken into account. The most relevant datasets have been selected in a subsequent stage.

Unfortunately, there is no a complete catalogue of the datasets published on the Web in machine-readable format. Although some efforts have been made in order to generate comprehensive catalogues of the available datasets in the Linked Data cloud, they barely contain the majority of the Linked Data resources, as a large number of data providers publish data in the cloud but do not register metadata about their publications. CKAN (Comprehensive Knowledge Archive Network) [51] is the most relevant catalogue of content in machine-readable format. Nevertheless, due to the nature of this registry, which relies on the willingness of the data providers to register metadata about their publications, it was expected that some datasets would not be found.

Therefore, a more comprehensive approach was conducted for the discovery of further datasets. The use of CKAN was complemented by the use of semantic search engines and browsers for the identification of prospective datasets.

Several Linked Data search engines have been developed to provide key-word search capabilities. The user interaction they offer is very similar, but the results returned can be considerably different. Thus, the discovery of datasets has been performed with the search of key words belonging to the domain of interest using different search engines. The key words correspond to the scientific names of species contained in the base dataset.

Additionally, semantic browsers were used to try to discover more datasets that were not found with the use of the previous tools. In this regard, a number of well-known Linked Data browsers are available, such as Marbles and Q&D RDF Browser, for crawling the Web of Data.

Thus, the discovery of potential datasets containing data of the domain of interest has been performed with the use of catalogues, search engines and browsers.

3.2.1. Use of Linked Data Catalogues

The CKAN catalogue is the most cited and visible Linked Data catalogue on the Web, and can be

found with a simple Google search of the string “linked data catalogue”. This catalogue is described in its web site as a registry of open knowledge packages and projects published in “formats that are machine automatable” [51]. Interestingly, this definition does not include the Linked Data or RDF concepts; instead, a broader term is used to describe it. Thus, it may be expected that datasets in different formats, including the Linked Data ones, can be found in this collection. The key feature of this catalogue is that anyone can register a new data package. As a result, this is the most comprehensive catalogue of Linked Data datasets, and is used for the generation of the LOD cloud diagram.

Two methods were used to find datasets in the catalogue. The first comprises a key word or phrase search that looks for datasets containing the given word or phrase in the metadata of the datasets. The second method consists in browsing a list of datasets published in the LOD Cloud.

By using the first approach with the word “fish” as key word, 18 packages were found. However, only one of them is actually related to fish species (Fishes of Texas). The rest of them are not relevant for our requirements, such as the “Fish Subsidy” and “Statistics of Stonia” data sources. Furthermore, some are provided in non-Linked Data formats; for example, spreadsheets or Esri shapefiles (a data format used for geographic information systems).

The list of packages in the LOD cloud provides the name and a brief description of the datasets published. The following datasets that may contain fish species data in RDF format were identified in the list:

- BBC Wildlife Finder
- Uniprot Taxonomy
- DBpedia
- EUNIS (European Nature Information System)
- Geospecies Knowledge Base
- Fishes of Texas (also found with the key word search).

A couple of other Linked Data catalogues can be found on the Web, but they are outdated and refer to the CKAN catalogue for an updated list of datasets. This is the case of the collection of RDF datasets published by the W3C project Linking Open Data on the Semantic Web [52].

3.2.2. Use of Linked Data Search Engines

A number of semantic search engines are available on the Web, the most popular include Swoogle, FalconS, WATSON, Sindice, Sig.ma and SWSE. Such search engines are used in a similar way to traditional web search engines, based on keyword or phrase searches, returning a list of resources. However, as Hogan et al. [53] point out, the results correspond to representations of real world things, not HTML documents. Hogan et al. also explain that semantic search engines work in an analogous way to traditional search engines, crawling, ranking and indexing data on the Web, but they contain special components for RDF data manipulation.

The selected procedure to find datasets containing fish species involves using different search engines to search for scientific names of species. If a given species is found, the dataset containing data about that particular fish species is examined to verify if it also contains data about more species.

In order to increase the probability of finding relevant datasets, the names of both popular and rare marine and freshwater fish species selected randomly have been used. Those names are *Sphyrna barracuda* (great barracuda), *Betta splendens* (Siamese fighting fish) and *Salmo salar* (Atlantic salmon) as names of common species, and *Bregmaceros bathymaster* (codlet) and *Apletodon dentatus* (Small-headed clingfish) as names of rare species.

Data of popular fish species is more likely to be published and found on the Web than that of rare species. However, it is very likely that data about common fish species is contained in publications about food, trade or other topics that may not provide relevant data to be linked to the FishDelish dataset. On the other hand, data about rare species is less likely to be found, but if it is, it is highly possible that the datasets containing the rare species also contain more valuable data suitable to be linked to the FishDelish dataset. As a result, a combination of popular and rare species can lead to the discovery of a satisfactory number of relevant datasets.

Huge differences among the effectiveness of the different search engines were found. For example, searching for the string *Sphyrna barracuda* using WATSON search engine returned 0 results; Swoogle returned 9 results but all belonged to the same source (Spire); while Sindice returned 61 results from different resources.

As expected, many results provided by the search engines were related with datasets that did not contain biological data of fish species, but other types of information about species. As an example,

the string *Salmo salar* returned 579 results using Sindice, but the vast majority of them were RDF documents from the Mendeley database (a database of scientific publications) containing metadata about papers related to this species, such as the RDF document “Identification of the sex-determining locus of Atlantic salmon (*Salmo salar*) on chromosome 2”.

After analysing the different results provided by the search engines, and selecting only those results that actually refer to datasets containing useful fish species data, the following summary was created, which presents the datasets that were found with each search engine:

Search Engine	Found dataset	String searched	Found URI
Swoogle (http://swoogle.umbc.edu)	Sipire	<i>Sphyræna barracuda</i>	http://spire.umbc.edu/ontologies/EthanAnimals.owl#Sphyræna_barracuda
Falcons (http://ws.nju.edu.cn/falcons)	DBpedia	<i>Betta splendens</i>	http://dbpedia.org/page/Siamese_fighting_fish
	BBC Wildlife	<i>Salmo salar</i>	http://www.bbc.co.uk/nature/life/Atlantic_salmon#species
	OpenCyc	<i>Betta splendens</i>	http://sw.opencyc.org/concept/Mx4rvw9w15wpEbGdrcN5Y29ycA
WATSON (http://watson.kmi.open.ac.uk)	no datasets containing fish species were found		
Sindice (http://sindice.com)	DBpedia	<i>Sphyræna barracuda</i>	http://dbpedia.org/page/Great_barracuda
	OpenCyc	<i>Betta splendens</i>	http://sw.opencyc.org/concept/Mx4rvhmcUJwpEbGdrcN5Y29ycA
	Taxonconcept	<i>Betta splendens</i>	http://lod.taxonconcept.org/ses/bGPhI.rdf
	Uniprot	<i>Apletodon dentatus</i>	http://www.uniprot.org/taxonomy/206117
Sig.ma (http://sig.ma)	DBpedia	<i>Sphyræna barracuda</i>	http://dbpedia.org/page/Great_barracuda
	Freebase	<i>Sphyræna barracuda</i>	http://www.freebase.com/view/wikipedia/images/en_id/891270
	Uniprot	<i>Apletodon dentatus</i>	http://purl.uniprot.org/taxonomy/206117
	Taxonconcept	<i>Betta splendens</i>	http://lod.taxonconcept.org/ses/bGPhI.rdf

Table 1: List of datasets found with the use of search engines

Several conclusions can be drawn from this exercise. Firstly, the search of popular fish species, such

as *Betta splendens*, led to the discovery of more datasets, as expected. However, the use of a rare species names was not futile as it allowed the discovery of one more dataset, Uniprot, which was found only with the search of the name *Apletodon dentatus*.

Secondly, the use of a single search engine is not sufficient for comprehensive results. The use of several semantic search engines is the best approach to find any kind of information in the Web of Data, as the number of results and resources that they can provide differs greatly from one search engine to another. Nevertheless, it is clear that some search engines are more effective, while others offer poor performance.

3.2.3. Use of Linked Data Browsers.

In this procedure, a number of semantic browsers have been used to find further fish species' data. Some of the URIs of the species that have been discovered in previous steps are used as starting point for the navigation of the Web of Data, with the purpose of identifying assertions such as *owl:sameAs* or *skos:exactMatch* pointing to entities from other resources. In other words, it is expected that some of the datasets that have been found contain RDF links to other datasets with fish species data.

Marbles and Q&D RDF Browser have been used, as they provide a simple and friendly interface and good performance for our purposes. In this exercise, unlike the search engines' case, we do not expect to receive different results according to the specific tool used. The Linked Data browsers only display the RDF data and links contained in the selected resource, independently of the specific browser being used, just as their counterparts in the traditional Web, that display the same HTML document, regardless of the used browser.

After browsing the Web of Data, it was found that the Taxonconcept dataset contains a number of links to other resources containing species (Uniprot, DBpedia, Freebase and Eunis). Most of these resources have already been discovered. However, two more datasets were found while navigating the Taxonconcept dataset using the Q&D RDF Browser: bio2rdf and Globalnames. The links between Taxonconcept and the other datasets are constructed using the *skos:closeMatch* predicate. It is worth noting that the number of links to other datasets in Taxonconcept depends on each particular species. While some species do not contain links at all, some contain links to all of the mentioned datasets.

3.2.4. Analysis of Results

A number of datasets were found using the different procedures previously discussed. The following table summarises the basic information about the found resources. Three types of datasets are identified: the first corresponds to datasets containing data from different fields of study, which is the case of DBpedia and Freebase; the second type corresponds to resources containing biological data from different topics, such as Uniprot and EUNIS, which not only contain species information but also data about proteins, habitats and other topics; and the third comprises datasets containing exclusively data of biological species, with Geospecies and Taxonconcept as examples.

Dataset	URL	Type	Method of identification
BBC Wildlife Finder	http://www.bbc.co.uk/nature/	Biology	catalogue search engine
Uniprot	http://www.uniprot.org	Biology	catalogue search engine semantic browser
DBpedia	http://dbpedia.org/	General knowledge	catalogue search engine semantic browser
EUNIS (European Nature Information System)	http://eunis.eea.europa.eu	Biology	catalogue semantic browser
Geospecies	http://lod.geospecies.org/	Species	catalogue
Fishes of Texas	http://www.fishesoftexas.org	Species	catalogue
Spire		Biology	search engine
OpenCyc	http://sw.opencyc.org/	General knowledge	search engine
Taxonconcept	http://www.taxonconcept.org/	Species	search engine
Freebase	http://www.freebase.com/	General knowledge	search engine semantic browser
bio2rdf	http://bio2rdf.org/	Biology	semantic browser
Globalnames	http://gni.globalnames.org/	Species	semantic browser

Table 2: Summary of the datasets found with each mechanism

The most effective method was the use of different semantic search engines combined with adequate key words from the domain of interest, which allowed the discovery of 8 of the 12 datasets. However, it can be concluded that the most convenient approach to finding datasets containing data from a given domain of interest is the use of different tools and procedures, as some are much more visible than others. This is the case of DBpedia, perhaps the most popular Linked Data resource, and Uniprot, which were identified with the three different methods used in this project. Some datasets such as Spire, Fishes of Texas and Taxonconcept were found with the use of only one tool.

3.3. Analysis and Selection of Datasets

3.3.1. Methodology

The next stage in this project involves analysing the datasets found in the previous phase to decide which are the most convenient to be linked to the FishDelish data source, while considering a number of factors.

An overview of each dataset is provided, accompanied by a description of the mechanisms used to extract data and calculate the three factors required for the selection of the most adequate datasets. These are the number of overlapping entities with the base dataset, accessibility to the data and an estimation of the number of assertions per species.

Once the value of each criterion was calculated for the different alternatives, a decision matrix was created to identify which datasets better meet our requirements.

Before presenting the analysis of each dataset, a brief description about these factors and how they are calculated is presented.

Accessibility to the data

This is evaluated by considering the existence or lack of means of access to the data, such as SPARQL endpoints and dump files. As it was previously mentioned in the background section, a common problem in the construction of links to external resources is the lack of mechanisms to query and retrieve data from the target data sources. In this respect, a resource that contains a large amount of fish species data can be useless for our purposes if the site does not provide any mechanisms to extract data efficiently.

Another important factor is the availability of documentation describing the structure of the triples contained in the RDF repository, as the construction of a SPARQL query requires the knowledge about the predicates and namespaces used in the assertions. That information can be extracted with the execution of a number of SPARQL queries designed for this purpose. However, the documentation facilitates the access to the required data to a high degree .

To evaluate this factor, one point is given to a dataset if it offers a SPARQL endpoint. Another point is awarded if it offers dump files containing the data in RDF format. One more point is given if

documentation describing the structure of the triples and the used vocabularies are provided, as well as some SPARQL queries as examples, or any other information describing the mechanisms offered to find and retrieve data.

If the number of results that can be retrieved via SPARQL queries is not restricted, another point is awarded. The reason for this is that the number of results returned by some SPARQL endpoints is limited to a small fixed number, usually around 1,000. The problem with this limitation is that commonly the Linked Data datasets contain a large amount of data. This restriction makes retrieving a large number of triples via SPARQL queries difficult, as several queries must be executed to extract the required data.

Finally, one more point is awarded if the site offers other means to access or download the data, such as CSV or Excel files containing the required information (URIs, scientific names and authority).

Number of overlaps

This involves counting the number of overlapping fish species between the target and base datasets. Different procedures could be performed to obtain this number. A simple approach is that, for each entity in the base dataset, generate manually a SPARQL query to retrieve the element in the target dataset and count the number of elements found. However, a manual approach is not efficient in situations where the number of entities is large, as in this case. Thus, the following semi-automated approach has been used to acquire the number of overlapping species for each prospective dataset:

- Extract a list containing all the URIs and scientific names of the species in the base dataset with a SPARQL query, and save this information in a file.
- Extract a list containing all the URIs and scientific names of the species in the target dataset with a SPARQL query, saving the information in a file. This query can be constructed easily for the datasets containing only fish species data. However, many of the datasets analysed contain not only fish species' data, but also other species' assertions. In the latter situation, the complete list of all the entities in the dataset is extracted if its size is not too large to be downloaded and manipulated. However, if the retrieval of all the species of the dataset represents technical difficulties because of its size, a SPARQL query that filters the fish species according to their taxonomic classification is generated. This last method is used in cases where the taxonomic classification system used in the dataset is known.
- Count the number of overlapping elements in both lists with the use of the Dataset

Comparison Application. As already discussed, this software parses the files containing the list of scientific names and performs a literal string comparisons of strings to acquire the number of overlaps.

This factor is particularly important because it represents the number of links that can be constructed.

Number of assertions per species

A desirable factor to consider in the evaluation of the datasets is the quality of the data contained in the datasets. However, computer scientists can hardly assess the value and accuracy of the data from the biological domain. Moreover, the design and implementation of an artefact to perform this evaluation would be complex given that quality may depend on both objective and subjective judgements.

Nevertheless, we can estimate the amount of information provided about the species contained in the datasets. This estimation is performed by obtaining the average number of assertions of a sample of fish species entities, for each of the potential datasets. Counting the number of triples of the datasets with a SPARQL query may not reflect the amount of relevant data contained in the datasets about fish, because some of the triples may correspond to assertions that are not related with it. On the other hand, the development of software to perform this task is not required as it only needs estimations, not accurate numbers.

To estimate the number for assertions per species for each dataset, a group of 10 entities have been selected randomly and the number of RDF triples that each of them contain was counted with the aid of the Q&D RDF browser, which provides this number for each entered URI. Then, the average number of triples was calculated. Finally, the dataset with the larger number of average assertions received 5 points for this factor, and the remaining datasets received a score calculated according to this scale.

3.3.2. BBC Wildlife Finder

In an effort to make available wildlife programmes online, the BBC created this site, which combines clips, pictures and information about wildlife, including data about animals, habitats and behaviours, taken from different resources such as Wikipedia [54]. The data is provided in two formats: standard HTML pages and as Linked Data in RDF format, offering a URI for each species. The BBC uses

content negotiation, providing either HTML or RDF data according to the client's request headers.

BBC has been interested largely in the Semantic Web as a technology to provide integration among its domain specific sites, allowing the users to discover further content and to navigate easily across the different BBC sites, improving the user experience [28]. BBC Programmes and BBC Music are other sites where content is offered as Linked Data in addition to HTML content.

Accessibility to the data

One SPARQL endpoint (<http://api.talis.com/stores/bbc-wildlife/services/sparql>) was found using Google with the phrase “bbc wildlife sparql endpoint” as there is no a direct link in the website. Nevertheless, the information that can be retrieved from the endpoint is not synchronised with the data that actually appears on the site, as it is demonstrated later. Therefore, only half a point related with the presence of a SPARQL endpoint is awarded. On the other hand, the number of results that can be retrieved using this endpoint is not limited, as the 23,861 assertions contained in this dataset can be extracted with a single query.

Dump files are not offered, or additional ways to retrieve data, such as CSV files; only a standard search textbox is provided. One point is awarded for the presence of some documentation with example queries (<http://blog.dbtune.org/post/2009/06/11/BBC-SPARQL-end-points> and <http://blog.dbtune.org/post/2009/06/15/And-another-fun-BBC-SPARQL-query>), which are supplied not by the BBC site but by external resources.

Concept	Comment	Points
SPARQL endpoint	Yes, but data not synchronised with the website.	0.5
Number of results limited	No	1
Documentation	Yes	1
Dump files with RDF data	No	0
Other ways to retrieve data	No	0
TOTAL		2.5

Table 3: BBC Wildlife Finder data accessibility

The score for the accessibility to the data in this dataset is presented in the table above.

Number of overlaps


In the main web page of this site (<http://www.bbc.co.uk/nature/wildlife>) the number of entities

contained in this data source is presented: 919 animals, 107 behaviours and 59 habitats. Of the 919 animals, only 39 correspond to fish, as evidenced in the figure below.

Explore:		Animals (919)	Behaviours (107)	Habitats (59)
Mammals (330)	Reptiles (115)	Insects (69)	Amphibians (24)	
Birds (268)	Plants (52)	Fungus (3)	Fish (39)	


Prehistoric animals	History of life on Earth	Dinosaurs
---------------------	--------------------------	-----------

Seaside spectacular



When it comes to summer holidays, there's no better place than the seaside and if you know where to look you'll be surprised at the wildlife you can find.

What's new?




Steller sea lion
▶ new clip


Places

Find wildlife


Most popular video clips



Tiny but deadly
Dart frogs might be small, but they're some of the planet's most poisonous animals.



Bird monument
The Farne Islands' Pinnacle Rocks are a birders paradise.



Immortal combat
The story of a battle frozen in time between Protoceratops and Velociraptor.

Figure 7: BBC Wildlife Finder number of entities
(<http://www.bbc.co.uk/nature/wildlife>)

A SPARQL query was constructed to confirm the number of fish species contained in this dataset. This query was created based on the fact that the BBC dataset contains data from fish species belonging to four taxonomic clades: *Chondrichthyes* (cartilaginous fish), *Actinopterygii* (ray-finned fishes), *Sarcopterygii* (lobe-finned fishes) and *Cephalaspidomorphi* (lampreys), according to information on the website. In examining the data of some species it was found that those clades are considered taxonomic classes in this dataset, thus resulting in the following SPARQL query.

```

SELECT count(?a)
WHERE
{
  {?a
    <http://purl.org/ontology/wo/class>
    <http://www.bbc.co.uk/nature/life/Actinopterygii#class> }
  UNION
  {?a
    <http://purl.org/ontology/wo/class>
    <http://www.bbc.co.uk/nature/life/Cephalaspidomorphi#class> }
  UNION
  {?a
    <http://purl.org/ontology/wo/class>
    <http://www.bbc.co.uk/nature/life/Chondrichthyes#class> }
  UNION
  {?a
    <http://purl.org/ontology/wo/class>
    <http://www.bbc.co.uk/nature/life/Sarcopterygii#class> }
}

```

The result returned by this query was 0, which evidently does not match the number of entities published in the site (39). There are two possible reasons for this mismatch. Firstly, the query is not constructed properly, because the RDF data of fish does not contain the assertion used in the query, or because the URIs used are incorrect. The second reason is that the data associated to the SPARQL endpoint does not correspond exactly with the data presented in the site. After analysing the data of some fish entities selected randomly, it was noted that all of them contain the assertion specifying their taxonomic class, and the URIs are correct. Thus, it was concluded that the data provided by the SPARQL endpoint is not synchronised with the data presented in the BBC Wildlife site.

The number of overlaps was counted manually as it contains only 39 fish entities that can be easily searched in the FishDelish dataset. An automatic approach involves the extraction of the data by other means considering that the SPARQL endpoint does not contain updated data, and the development of Java code to parse the file containing the data. For the previous reasons, a manual counting was more convenient, with 16 overlaps found.

Number of assertions per species

The list of species selected randomly and the number of triples contained in each is presented in the following table:

Species	URI	No. of assertions
<i>Rhincodon typus</i> (whale shark)	http://www.bbc.co.uk/nature/life/Whale_shark	114
<i>Istiophorus albicans</i> (Atlantic sailfish)	http://www.bbc.co.uk/nature/life/Atlantic_sailfish	102
<i>Engraulis ringens</i> (Peruvian anchovy)	http://www.bbc.co.uk/nature/life/Peruvian_anchoveta	94
<i>Arapaima gigas</i> (Arapaima)	http://www.bbc.co.uk/nature/life/Arapaima	93
<i>Galeocerdo cuvier</i> (tiger shark)	http://www.bbc.co.uk/nature/life/Tiger_shark	109
<i>Salmo salar</i> (Atlantic salmon)	http://www.bbc.co.uk/nature/life/Atlantic_salmon	137
<i>Phyllopteryx taeniolatus</i> (Weedy sea dragon)	http://www.bbc.co.uk/nature/life/Phyllopteryx	92
<i>Lampetra planeri</i> (European brook lamprey)	http://www.bbc.co.uk/nature/life/European_brook_lamprey	91
<i>Manta birostris</i> (Atlantic manta)	http://www.bbc.co.uk/nature/life/Manta_ray	106
<i>Pygocentrus nattereri</i> (red-bellied piranha)	http://www.bbc.co.uk/nature/life/Red-bellied_piranha	112
Average	105	

Table 4: BBC Wildlife Finder estimated number of triples per species

3.3.3. Uniprot Taxonomy

The Uniprot Knowledgebase is a collection of databases related to proteins, including data about amino acid sequences, taxonomic data and citation information [55]. Its main objective is to provide a comprehensive and annotated database of protein sequences to the scientific community, encouraging biological research. This knowledge base is generated, curated and maintained by the Uniprot Consortium, and integrates data from several resources, acting as a hub for biomolecular data by linking more than 140 databases [56]. It is a comprehensive, updated and reliable resource of biological information.

One of the datasets that comprise the Uniprot Knowledgebase is the Uniprot Taxonomy database, which contains information on 938,654 organisms, which are classified in a hierarchical tree structure. According to information on the Uniprot website [55], the taxonomy data is curated manually and the scientific names of the organisms are manually verified. The binominal system (genus and species names) for scientific names is used in this database for the organism denomination.

Accessibility to the data

Several methods of obtaining information are available in the Uniprot Taxonomy website. The first way is a keyword search, where the string typed by the user is looked up in the metadata of the organisms. An advanced search interface is also provided, where the user can select the specific field (scientific name, common name, rank, ID, etc.) to look up the required string.

A second mechanism consists of an interface to browse the data according to the hierarchical classification of the organisms. Additionally, the complete list containing the 938,654 elements can be browsed, with the possibility of selecting the number of results displayed by page, from 10 to 250.

Independently of the mechanisms used to find and select organisms, the list of results can be downloaded in four different formats: tab-delimited file, MS Excel file, RDF/XML format, and as a list in a plain text file. Moreover, the data about each specific organism is displayed in a web page which contains a link to the RDF version.

A rich amount of documentation is provided in the website, including tutorials, guides and other documents describing how to retrieve data using the different mechanisms available in the site. As examples of this documentation, a detailed description of the key word search is offered at <http://www.uniprot.org/help/text-search>, and a guide to retrieve data from different elements in a

single operation is provided in the URL <http://www.uniprot.org/help/batch>.

An SPARQL endpoint is available at <http://uniprot.bio2rdf.org/sparql>. The number of triples than can be retrieved via this endpoint is limited to 10,000, which is not a big restriction. Even when the Uniprot website does not contain documentation and examples about SPARQL queries, some examples and further information can be found on the web (<http://hublog.hubmed.org/archives/001789.html>, <http://www.w3.org/wiki/LifeSciencesQueries/SPARQL>).

Thus, five points are awarded to this dataset for the accessibility to its data, as it offers a SPARQL endpoint and other methods of accessing and downloading data. Additionally, some documentation can be found on the Web and, even when the SPARQL is limited to 10,000 results, this number is big enough to not hinder the retrieving of data, therefore half a point is given. Finally, dump files can be downloaded containing RDF data.

Concept	Comment	Points
SPARQL endpoint	Yes	1
Number of results limited	10000	0.5
Documentation	Yes	1
Dump files with RDF data	Yes	1
Other ways to retrieve data	Yes (Excel, Tab-delimited and plain text files)	1
TOTAL		4.5

Table 5: Uniprot data accessibility

Number of overlaps

Using the hierarchical navigation, the taxonomic group of chordates (phylum *chordata*) was downloaded in RDF/XML format. It is known that all fish species are classified inside this phylum. Thus, a single file, which could be easily downloaded without the need of writing SPARQL queries, contains data of all the fish species in the Uniprot database. This file has data of 55,969 entities (as of 17th of May 2011).

One more class was developed for the Dataset Comparison Application, with the purpose of converting the RDF file in an array of objects of the class *Species* previously described. After executing the application, the number of species with the same scientific name in both datasets was 10,541.

Number of assertions per species

The following table shows the estimated number of assertions that this dataset contains for each entity.

Species	URI	No. of assertions
<i>Carcharias taurus</i> (Sand tiger shark)	http://purl.uniprot.org/taxonomy/30501	31
<i>Cyprinella callitaenia</i> (bluestripe shiner)	http://purl.uniprot.org/taxonomy/87711	10
<i>Dentex maroccanus</i> (Morocco dentex)	http://purl.uniprot.org/taxonomy/98815	10
<i>Epinephelus tauvina</i> (greasy grouper)	http://purl.uniprot.org/taxonomy/203262	10
<i>Labidochromis gigas</i>	http://purl.uniprot.org/taxonomy/445512	11
<i>Lutjanus argentimaculatus</i> (mangrove red snapper)	http://purl.uniprot.org/taxonomy/211834	10
<i>Nansenia candida</i> (bluethroat argentine)	http://purl.uniprot.org/taxonomy/557352	10
<i>Pangio pangia</i>	http://purl.uniprot.org/taxonomy/457500	9
<i>Rasbora caverii</i> (Cauvery rasbora)	http://purl.uniprot.org/taxonomy/244131	11
<i>Sebastes elongatus</i> (greenstriped rockfish)	http://purl.uniprot.org/taxonomy/72070	10
Average	12.2	

Table 6: Uniprot estimated number of triples per species

3.3.4. EUNIS (European Nature Information System)

The EUNIS biodiversity database is one of the databases created and maintained by the European Environment Agency (EEA). The EEA is an agency of the European Union whose main purpose is to provide information on the environment to its member countries in order to support them in making the right decisions concerning the preservation of the environment, sustainability and the integration of environmental concerns into economic policies [57].

This database contains information about species, habitat types and sites from Europe. The species' section contains data about more than 275,000 taxonomic groups from different ranks that are present in Europe [57]. A SPARQL query to count the number of triples contained in this dataset, which is presented below, returns the number 20,229,105 (July 2011).


```
SELECT count(?a) WHERE {
    ?a ?b ?c
}
```

Accessibility to the data

Several ways of finding information are offered in the EUNIS website. A hierarchical navigation of species according to their taxonomic classification is one of the tools. Additionally, data can be found using a classic keyword search interface. An advanced search tool is provided, which allows the construction of more specific queries. There are some predefined search functions which perform the most frequently used queries, such as species by common groups (mammals, fish, amphibians), located within a country, or threatened species. In the web pages containing datasets, a link is provided to download the data in different formats, including TSV and XML. However, no RDF dump files are offered.

Comprehensive and clear documentation describing the different methods of finding and retrieving data is available on this site. Moreover, some interesting animated tutorials developed in Adobe Flash provide guidance on the use of the EUNIS database in a highly attractive and understandable design, explaining step by step how to perform some common operation such as the search of species by name, region or by taxonomic groups.

A SPARQL endpoint is available at <http://cr3.eionet.europa.eu/sparql>. The number of results that can be returned is limited to 2,000 items. No examples of SPARQL queries were found in the EUNIS website or in the rest of the Web using search engines. The summary of the points awarded by this site regarding the ease of access its data is presented in the following table.

Concept	Comment	Points
SPARQL endpoint	Yes	1
Number of results limited	2000	0
Documentation	Yes	1
Dump files with RDF data	No	0
Other ways to retrieve data	Yes (XML and tab-separated files)	1
TOTAL		3

Table 7: EUNIS data accessibility

Number of overlaps

The number of fish species contained in the EUNIS database is 2,686. This number was found using one of the predefined searches to find groups of species. The figure below shows the web interface

used to obtain this information.

Species group

[Download results](#)

You searched species from group **Fishes**

Results found: **2686**

Results displayed per page (max. 300) [Change](#)

Refine your search

Scientific name is [Search](#)

Current page: 1 / 269

Go to page: [Change page](#)

Group	Order	Family	Scientific name
Fishes		Acipenseridae	Acipenser naccarii
Fishes		Acipenseridae	Acipenser sturio
Fishes		Cyprinidae	Alburnus albidus
Fishes		Clupeidae	Alosa alosa
Fishes		Clupeidae	Alosa fallax
Fishes		Clupeidae	Alosa macedonica
Fishes		Cyprinidae	Anaecypris hispanica
Fishes		Anguillidae	Anguilla anguilla
Fishes		Cyprinodontidae	Aphanius fasciatus
Fishes		Cyprinodontidae	Aphanius iberus
Group	Order	Family	Scientific name

Current page: 1 / 269

Go to page: [Change page](#)

Figure 8: EUNIS number of fish species entities

([http://eunis.eea.europa.eu/species-groups-result.jsp?](http://eunis.eea.europa.eu/species-groups-result.jsp?showScientificName=true&expand=false&showGroup=true&showOrder=true&showFamily=true&groupID=2&groupName=Fishes&submit=Search)

[showScientificName=true&expand=false&showGroup=true&showOrder=true&showFamily=true&groupID=2&groupName=Fishes&submit=Search](http://eunis.eea.europa.eu/species-groups-result.jsp?showScientificName=true&expand=false&showGroup=true&showOrder=true&showFamily=true&groupID=2&groupName=Fishes&submit=Search))

An XML file with a list of all fish species in the dataset was downloaded from the web page in the last figure. Then, a new class was added to the Dataset Comparison Application to parse the XML file and convert it in a Java list. After executing the application it was found that the number of overlapping species between the EUNIS and FishDelish datasets is 2,250.

Number of assertions per species

The next table shows the estimated number of assertions per species.

Species	URI	No. of assertions
<i>Acipenser sturio</i> (Atlantic sturgeon)	http://eunis.eea.europa.eu/species/397	225
<i>Chondrostoma genei</i> (South European nase)	http://eunis.eea.europa.eu/species/451	186
<i>Lethenteron zanandreaei</i> (Lombardy brook lamprey)	http://eunis.eea.europa.eu/species/530	139
<i>Rutilus alburnoides</i> (calandino)	http://eunis.eea.europa.eu/species/583	230
<i>Carcharodon carcharias</i> (Great white shark)	http://eunis.eea.europa.eu/species/8650	112
<i>Ambloplites rupestris</i> (Rock bass)	http://eunis.eea.europa.eu/species/10068	33
<i>Salaria fluviatilis</i> (freshwater blenny)	http://eunis.eea.europa.eu/species/10089	82
<i>Nemacheilus angorae</i> (angora loach)	http://eunis.eea.europa.eu/species/10237	26
<i>Bonapartia pedaliota</i>	http://eunis.eea.europa.eu/species/125310	22
<i>Alepisaurus brevirostris</i> (lancet fish)	http://eunis.eea.europa.eu/species/124906	29
Average	108.4	

Table 8: EUNIS estimated number of triples per species

3.3.5. Geospecies

The Geospecies Knowledge Base was created to link dispersed data about species, such as genetic and morphological characteristics, weather, locations, time and other conditions of observations, climatological and other attributes of locations, etc. with the use of Linked Data [27].

Its main goal is to create unique identifiers for species that remain stable despite changes in their taxonomic classification or in their scientific name. With this approach, a robust link joins species' occurrence records, scientific articles, genetic information, etc., because changes in nomenclature or taxonomy will not affect those links [27].

The dataset contains data about plants and animals from all over the world. A URI is assigned to each species, which resolves a HTML web page or RDF document depending on the type of client. According to information provided in this site, the database contains data about 19,230 species, and the total number of triples in the dataset is 2,201,532 (April 2010 version). One of the relevant aspects of this dataset is that it contains links to other resources. Of the 19,230 species, 11,799 are linked to DBpedia, 11,095 are linked to bio2rdf and Uniprot, and 2,676 to the EUNIS database using the predicate *skos:closeMatch* [27].

The site is hosted by the Department of Entomology of the University of Wisconsin.

Accessibility to the data

There are three ways to retrieve data in this site. The first involves a web interface allowing a hierarchical navigation in the taxonomic tree. In order to find a specific species it is necessary to know its taxonomic classification. No keyword search or other tools to find data are offered.

The second way to retrieve data is a SPARQL endpoint available at <http://lod.openlinksw.com/sparql>, which is limited to 100,000 results. Finally, a dump file containing all the RDF triples in this dataset can be downloaded.

The amount of documentation about the dataset and the mechanisms to retrieve data is limited but, at the same time, the provided information is relevant and useful. There is one web page with some interesting SPARQL query examples (list of Bats in the UK, species with observation records in Door County, Wisconsin, species that have a page in The BBC Wildlife dataset, etc.).

The number of points awarded by this dataset regarding the accessibility to data is presented in the

following table.

Concept	Comment	Points
SPARQL endpoint	Yes	1
Number of results limited	100000	1
Documentation	Yes	1
Dump files with RDF data	Yes	1
Other ways to retrieve data	No	0
TOTAL		4

Table 9: Geospecies data accessibility

Once again, even when the number of results that the SPARQL endpoint can return is restricted to 100,000, this limitation does not represent a drawback for our purposes, as the number of records that might be retrieved is fewer than 100,000. Thus, one point is given for this concept.

Number of overlaps

The following SPARQL query was constructed to obtain the list of fish species contained in the Geospecies Knowledge Base.

```
PREFIX geospecies: <http://rdf.geospecies.org/ont/geospecies#>
SELECT ?species ?scientific_name
WHERE {
  {?species geospecies:hasScientificName ?scientific_name}
  {?species geospecies:hasClassName "Actinopterygii" }
  UNION
  {?species geospecies:hasClassName "Cephalaspidomorphi" }
  UNION
  {?species geospecies:hasClassName "Chondrichthyes" }
  UNION
  {?species geospecies:hasClassName "Elasmobranchii" }
  UNION
  {?species geospecies:hasClassName "Sarcopterygii" }
}
```

The previous query was designed to extract the URIs and scientific names of the species belonging to five different taxonomic classes. As discussed in the Classification of Fish section, some differences can be found in the taxonomic classification of fish species used in the different datasets. However, it can be confirmed that this query is adequate for this dataset by looking at the list of

classes grouped in the phylum *chordata* using the hierarchical navigation tree.

Home Animalia Phylum: Chordata Common Name: Chordates RDF												
<h2>Chordata</h2> <p>Chordates</p>												
Class	Common Name	uBio	NCBI	ITIS	GBIF	EOL	Wikipedia	Wikispecies	ToL	Google	Google	Google
Cephalochordata class	Lancelets									Books	Scholar	Text
Appendicularia	Larvacea									Books	Scholar	Text
Ascidacea	Sea Squirts									Books	Scholar	Text
Thaliacea	Salps									Books	Scholar	Text
Actinopterygii	Ray-finned Fishes									Books	Scholar	Text
Amphibia	Amphibians									Books	Scholar	Text
Aves	Birds									Books	Scholar	Text
Cephalaspidomorphi	Lampreys									Books	Scholar	Text
Chondrichthyes	Cartilaginous Fishes									Books	Scholar	Text
Elasmobranchii	Sharks, Rays									Books	Scholar	Text
Mammalia	Mammals									Books	Scholar	Text
Reptilia	Reptiles									Books	Scholar	Text
Sarcopterygii	Lobe-finned Fishes									Books	Scholar	Text

Figure 9: Phylum chordata in Geospecies
(<http://lod.geospecies.org/phyla/Ac1.xhtml>)

As it can be appreciated in the figure, this dataset refers to the five classes of fish (*Actinopterygii*, *Cephalaspidomorphi*, *Chondrichthyes*, *Elasmobranchii* and *Sarcopterygii*) used in the SPARQL query.

The number of results returned by the query was 28. This number appears inaccurate given the fact that this dataset contains data of 19,230 species, according to information in the site. Nevertheless, a manual inspection using the hierarchical tree confirmed that the number of fish species that this dataset contains is actually 28.

Due to the small number of entities, a manual counting was used to get the number of overlaps, which is 28. This means that all the fish species in this dataset are contained in FishDelish.

Number of assertions per species

This estimation is presented in the following table.

Species	URI	No. of assertions
<i>Coregonus artedii</i>	http://lod.geospecies.org/ses/7Pp88	106
<i>Oncorhynchus gorbuscha</i>	http://lod.geospecies.org/ses/f7dhT	107
<i>Prosopium cylindraceum</i>	http://lod.geospecies.org/ses/ondoX	107

<i>Coregonus hoyi</i>	http://lod.geospecies.org/ses/Z56au	104
<i>Salvelinus fontinalis</i>	http://lod.geospecies.org/ses/DmLZA	107
<i>Salmo trutta</i>	http://lod.geospecies.org/ses/kFudm	112
<i>Oncorhynchus tshawytscha</i>	http://lod.geospecies.org/ses/2Zquc	113
<i>Coregonus nigripinnis</i>	http://lod.geospecies.org/ses/rSPDv	104
<i>Coregonus clupeaformis</i>	http://lod.geospecies.org/ses/m5XJQ	106
<i>Oncorhynchus kisutch</i>	http://lod.geospecies.org/ses/xRqX3	113
Average	107.9	

Table 10: Geospecies estimated number of triples per species

3.3.6. Fishes of Texas

This dataset was found in the CKAN catalogue which indicates that it is available in RDF/XML format. The main objective of this project is to “to compile and synthesize knowledge of the spatial and temporal distribution of Texas freshwater fish fauna” [58]. This database contains 20,931 records of fish species (22th of July 2011), and provides a large amount of documentation, maps, statistics and other information. However, the data is stored in a relational database, and no information about the use of semantic technologies was found on the whole site. The documentation home page of the project indicates that the content in this site is still in development and should be regarded as provisional. Moreover, information in the CKAN catalogue reveals that this project is not yet fully available. Hence, as the data in RDF format is not available yet, this dataset was not taken into account in the development of this dissertation.

3.3.7. Spire

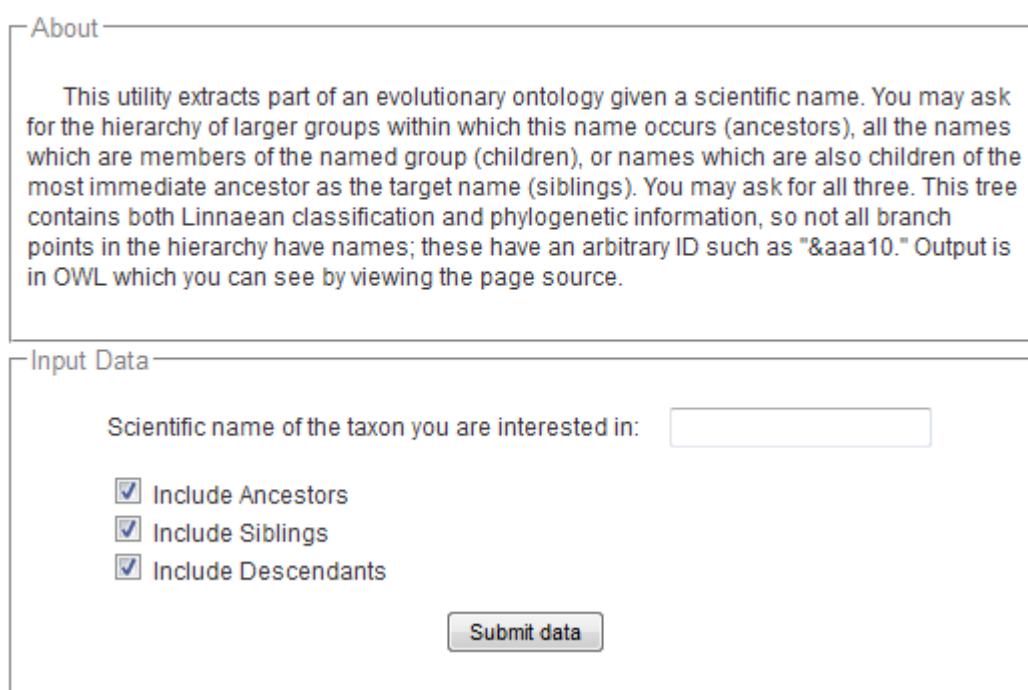
Spire is a research project created to investigate the use of semantic technologies to support science, specifically eco-informatics, by implementing and evaluating Semantic Web tools and applications for use in biological domains. Additionally, a framework to support research and education using the Semantic Web will be developed, comprising a set of ontologies, protocols, agents and other tools [59].

The documentation and information about the content of this resource and the tools to find data is poor; no information about the structure and number of records contained in the dataset is available. Unfortunately, this is another project which seems to be still under development or was created a long time ago and is no longer maintained, as the SPARQL endpoint does not work. Even the simple query that has been used to get the number of assertions in the dataset does not work, nor any of

the four example queries provided. The HTTP error code 500 (internal server error) is returned as the response to any query. No RDF dump files or any other kind of files containing data are available to download.

The only way to extract data of species is a web interface (presented in the figure 10) that, when given a scientific name, returns a set of RDF triples extracted from a natural history ontology called ETHAN. However, this tool is not useful for our purposes, as we require the automatic extraction of large amounts of data.

Spire: ETHAN Ontology



The screenshot shows a web interface for the Spire: ETHAN Ontology. It is divided into two main sections: 'About' and 'Input Data'. The 'About' section contains a paragraph explaining the utility's function: extracting parts of an evolutionary ontology based on a scientific name, providing hierarchy (ancestors), members (children), or siblings. It also mentions that the output is in OWL format. The 'Input Data' section features a text input field for a scientific name, three checked checkboxes labeled 'Include Ancestors', 'Include Siblings', and 'Include Descendants', and a 'Submit data' button.

About

This utility extracts part of an evolutionary ontology given a scientific name. You may ask for the hierarchy of larger groups within which this name occurs (ancestors), all the names which are members of the named group (children), or names which are also children of the most immediate ancestor as the target name (siblings). You may ask for all three. This tree contains both Linnaean classification and phylogenetic information, so not all branch points in the hierarchy have names; these have an arbitrary ID such as "&aaa10." Output is in OWL which you can see by viewing the page source.

Input Data

Scientific name of the taxon you are interested in:

☒ Include Ancestors
☒ Include Siblings
☒ Include Descendants

Figure 10: Spire search interface
(<http://spire.umbc.edu/ont/ethan.php>)

Due to the inability to explore the content of this dataset using the provided SPARQL endpoint and the lack of other ways to extract large amounts of data, the Spire dataset is not considered as a prospect dataset to be linked to the FishDelish dataset.

3.3.8. Taxonconcept

Pete DeVries, who created and maintains this dataset, is a specialist in the field of biodiversity informatics. He comments that the integration of different datasets containing species data including DNA sequences, images, proteins and occurrence records, facilitates the analysis and understanding of species information, and the semantic technologies offer a suitable set of techniques and tools to perform this integration, creating a useful unified knowledge base [50].

The species identifiers used in this database are not related to their taxonomic classification. The author recognises the limitations of binominal names as identifiers, as they lack the stability and uniqueness required to serve as adequate identifiers, as explained in the background section of this study.

This dataset contains 25,575,586 triples (24th of June, 2011) about species occurrences, including data of species, locations and details of the observation. Some of these correspond to links to the following datasets: BBC Wildlife, DBpedia, EUNIS, Geonames and Uniprot.

Accessibility to the data

Two SPARQL endpoints are provided (one powered by Openlink Virtuoso at <http://lsd.taxonconcept.org/sparql> and the other by OpenLink iSPARQL at <http://lsd.taxonconcept.org/isparql/>). The maximum number of results that can be returned using the first one is limited to 1,000,000, while the second is limited to only 50 results.

A number of RDF dump files containing different information (species, species occurrences, mappings, etc.) can be downloaded. Additionally, some useful example queries are offered in this site, and further information about this project is published in a blog. However, there are no other ways to find and download data, such as web interfaces or other types of files (i.e. CSV or Excel files).

Concept	Comment	Points
SPARQL endpoint	yes	1
Number of results limited	1000000	1
Documentation	yes	1
Dump files with RDF data	yes	1
Other ways to retrieve data	no	0
TOTAL		4

Table 11: Taxonconcept data accessibility

Thus, as can be seen from the table above, the number of points awarded by this dataset regarding the ease of accessing data is 4.

Number of overlaps

The following query was constructed to retrieve a list containing all the species in the dataset. It was not possible to create a query to retrieve only the fish species, as the taxonomic classification used in this dataset is not provided.


```

PREFIX txn: <http://lod.taxonconcept.org/ontology/txn.owl#>
SELECT ?species, ?scientific_name
WHERE
{
    ?species txn:hasScientificName ?scientific_name.
}

```

Once the list of species in XML format was downloaded, which contained 108,024 species, a new class for the Dataset Comparison Application was developed to extract the data from that file and populate the Java data structure required to count the number of species that overlap between the Taxonconcept and FishDelish dataset. After making the required modifications to the application and executing it, it was found that the number of overlapping species is 6,844.

Number of assertions per species

The following table presents the analysis of this factor:

Species	URI	No. of assertions
Ctenacis fehlmanni	http://lod.taxonconcept.org/ses/5iUrV#Species	121
Squalius keadicus	http://lod.taxonconcept.org/ses/439pL#Species	128
Neolissochilus soroides	http://lod.taxonconcept.org/ses/9xpfN#Species	118
Psilorhynchus arunachalensis	http://lod.taxonconcept.org/ses/DXzGH#Species	118
Barbus zanzibaricus	http://lod.taxonconcept.org/ses/Pu3AZ#Species	123
Garra bispinosa	http://lod.taxonconcept.org/ses/VwqDK#Species	123
Sebastes nigrocinctus	http://lod.taxonconcept.org/ses/fz3kh#Species	131
Siganus vulpinus	http://lod.taxonconcept.org/ses/ma2dZ#Species	135
Gollum attenuatus	http://lod.taxonconcept.org/ses/rMVtn#Species	130
Moxostoma breviceps	http://lod.taxonconcept.org/ses/zPnAG#Species	123
Average	125	

Table 12: Taxonconcept estimated number of triples per species

3.3.9. DBpedia

DBpedia is the most popular and one of the biggest datasets in RDF format. Its main objective is to reuse the vast amount of information contained in Wikipedia and offer it in machine-readable format

in the Web of Data.

Wikipedia has become probably the most important resource of knowledge available on the Web that integrates information from all the fields of study in a single knowledge base. According to the Alexa Top Sites index [60], Wikipedia is the seventh most visited site on the Web (15th of July, 2011). It is written collaboratively by more than 82,000 active contributors who work on more than 17,000,000 articles in more than 270 languages [61].

One of the biggest advantages of Wikipedia is that its information is created and updated constantly. However, this flexibility can bring some inconveniences: inaccurate information, unencyclopedic content and vandalism [61]. There are only two ways to find data in Wikipedia, by keyword searches or content navigation.

DBpedia's main purpose is to use Semantic Web technologies to convert all the content from Wikipedia in structured knowledge to allow the construction of complex queries for the extraction of data, the creation of links to other datasets and the development of mashups, according to Auer et al. [30]. With DBpedia, answers to complex questions can be answered, such as "Which Rivers flow into the Rhine and are longer than 50 kilometres?" [29], or "Which Skyscrapers in China have more than 50 floors and have been constructed before the year 2000?" [29].

Wikipedia already contains structured information in the form of info-boxes, geo-coordinates, categorisation information and other components. This information can be translated easily into RDF format. However, most information in Wikipedia is stored as free text, requiring the use of special techniques to parse the article texts and extract structured data [30]. Auer et al. [30] also explain that semantic relationships are created by mapping those in the relational database tables into RDF assertions, and by extracting further data from the article texts and info-boxes.

In July 2011, the DBpedia dataset "describes more than 3.5 million things, out of which 1.67 million are classified in a consistent Ontology, including 364,000 persons, 462,000 places, 99,000 music albums, 54,000 films, 17,000 video games, 148,000 organisations, 169,000 species and 5,200 diseases. The DBpedia data set features labels and abstracts for these 3.5 million things in up to 97 different languages; 1,850,000 links to images and 5,900,000 links to external web pages; 6,500,000 external links into other RDF datasets, 633,000 Wikipedia categories, and 2,900,000 YAGO categories." [29]. According to this resource, DBpedia contains over 672,000,000 RDF triples, of which, 286,000,000 were extracted from the English edition and the remainder from editions in other

languages.

The DBpedia dataset is linked to a large number of other public datasets from different domains. As it can be appreciated in the LOD cloud diagram (Appendix A), DBpedia is the centre of the cloud, and is linked to many of the most popular Linked Data datasets, such as Freebase, Flickr Wrapper, Drug Bank, Music Brainz, Geonames and US Census.

Accessibility to the data

DBpedia provides different tools to find data in its dataset. One of them is the SPARQL endpoint available at <http://DBpedia.org/sparql>. A key word search web interface is also offered at <http://DBpedia.org/fct>, and a web service with the same functionality can be found at <http://DBpedia.org/fct/service>. Additionally, the DBpedia dataset can be downloaded as RDF dump files.

Thus, the number of points awarded to this dataset regarding the ease of data retrieval is presented in the following table.

Concept	Comment	Points
SPARQL endpoint	Yes	1
Number of results limited	2000	0
Documentation	Yes	1
Dump files with RDF data	Yes	1
Other ways to retrieve data	Yes	1
TOTAL		4

Table 13: DBpedia data accessibility

Number of overlaps

For this dataset it was necessary to construct a query to retrieve a list containing only the fish species for the following reasons.

The number of triples that can be retrieved via the SPARQL endpoint is limited to only 2,000 elements. Consequently, the generation of a list containing all the chordates or all the species that belong to a broader rank would require the execution of a large number of queries. Thus, it is necessary to filter the data as much as possible in order to reduce the number of required queries.

Secondly, the data of species is not standardised in this dataset. Some species may contain some assertions while others may not. In particular, a small number of fish species do not contain the

dbpedia-owl:kingdom or *dbpedia-owl:phylum* predicates. Consequently, a SPARQL query to retrieve the list of all the species of the chordata phylum or all the species belonging to the animal kingdom, may ignore those species.

The following query was used to get the list of fish species in this dataset, which was constructed according to the taxonomic classification of fishes used by Wikipedia. As it can be noticed, this classification is consistent with that presented by Hickman et al. [42], which consists of 5 classes of living fishes (*Actinopterygii*, *Chondrichthyes*, *Myxini*, *Petromyzontida* and *Sarcopterygii*).

```
PREFIX dbpprop: <http://dbpedia.org/property/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
SELECT ?element, ?binominal WHERE {
  ?element dbpprop:binomial ?binominal .
  {?element dbpedia-owl:class dbpedia:Actinopterygii}
  UNION
  {?element dbpedia-owl:class dbpedia:Cephalaspidomorphi}
  UNION
  {?element dbpedia-owl:class dbpedia:Chondrichthyes}
  UNION
  {?element dbpedia-owl:class dbpedia:Myxini}
  UNION
  {?element dbpedia-owl:class dbpedia:Petromyzontida}
  UNION
  {?element dbpedia-owl:class dbpedia:Placodermi}
  UNION
  {?element dbpedia-owl:class dbpedia:Sarcopterygii}
}
ORDER BY ?binominal
```

The taxonomic classification in Wikipedia also takes into account seven more classes of extinct primitive fishes (*Pteraspisomorphi*, *Thelodonti*, *Anaspida*, *Conodonts*, *Cephalaspidomorphi*, *Placodermi* and *Acanthodii*). However, the DBpedia dataset only contains data from two of these seven classes of extinct fishes (*Cephalaspidomorphi* and *Placodermi*), which were also considered in the previous query. The number of species contained in each class could be counted with the execution of a simple query, such as the one presented below.

```

PREFIX dbpprop: <http://dbpedia.org/property/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
SELECT count(?element) WHERE {
    ?element dbpprop:binomial ?bionominal .
    ?element dbpedia-owl:class dbpedia:Acanthodii
}

```

The query for all fish species returned 9,145 elements. Once again, a new class to convert the XML data into a Java data structure was developed. After applying some minor changes to the Dataset Comparison Application and executing it, it was found that the number of species that are contained in both FishDelish and DBpedia datasets is 8,151.

Number of assertions per species

The estimated number of assertions per species in Dbpedia is presented in the following table.

Species	URI	No. of assertions
<i>Toxotes microlepis</i> (smallscale archerfish)	http://dbpedia.org/resource/Smallscale_archerfish	36
<i>Puntius dorsimaculatus</i>	http://dbpedia.org/resource/Puntius_dorsimaculatus	32
<i>Pteroidichthys amboinensis</i> (Ambon scorpionfish)	http://dbpedia.org/resource/Ambon_scorpionfish	46
<i>Barbus parawaldron</i>	http://dbpedia.org/resource/Barbus_parawaldroni	32
<i>Labeo nunensis</i>	http://dbpedia.org/resource/Labeo_nunensis	32
<i>Xiphophorus alvarez</i> (Chiapas swordtail)	http://dbpedia.org/resource/Chiapas_swordtail	30
<i>Hyporhamphus ihi</i> (New Zealand piper)	http://dbpedia.org/resource/New_Zealand_piper	39
<i>Stegastes sanctipauli</i>	http://dbpedia.org/resource/Stegastes_sanctipauli	46
<i>Gobius cobitis</i> (giant goby)	http://dbpedia.org/resource/Giant_goby	46
<i>Schistura spekuli</i>	http://dbpedia.org/resource/Schistura_spekuli	31
Average	37	

Table 14: Dbpedia estimated number of triples per species

3.3.10. Freebase

Freebase is an open repository of structured data containing around 22,000,000 entities about people, places, films, music, sports and all kinds of subjects. The data comes from a large number of

diverse data sources, including Wikipedia, The World Factbook, databaseFootball.com , Stanford University and Nature [62].

Freebase shares some similarities with Wikipedia, as it is a collaborative project and a free source of information. However, the Freebase system does not run on a wiki software but on Linked Data technologies. Freebase differs due to the fact that the information in Wikipedia is provided through text articles in a standard web interface, while Freebase uses semantic technologies to structure its data [62].

The question that arises is how similar or different this dataset is compared to DBpedia. In the same site [62], the answer to this question is provided. The similarity with DBpedia is that both projects extract data from Wikipedia to structure it in RDF format. The differences lay in two facts. DBpedia extracts data only from Wikipedia, while Freebase imports data from a large number of resources. Moreover, DBpedia is funded by several organisations, while Freebase is funded and owned by Google.

Accessibility to the data

The most visible way to find data is a web interface where the user can navigate by categories arranged in a hierarchical way. Additionally, a keyword search textbox is present in all the pages of the website.

Freebase does not provide a SPARQL endpoint. Instead, it provides a MQL query editor (<http://www.freebase.com/queryeditor>). MQL is the required language to query the Freebase database, which is, according to information in this site, analogous to SPARQL, with the difference that it uses HTTP requests and JSON objects to retrieve data. A rich set of documentation about MQL can be found on this site, including manuals, tutorials and reference guides. Nevertheless, it seems strange that the language to get data from this dataset, which is structured as Linked Data, is MQL instead of SPARQL, especially when SPARQL is the standard language to query data from RDF repositories, while MQL is a language with a reduce presence and popularity in the world of semantic technologies. Consequently, a user wishing to extract data more efficiently from the Freebase dataset must learn MQL, and apart from the Freebase site, it is very difficult to find further documentation about this language in the web.

There are data dump files containing tab-separated text that can be downloaded from http://wiki.freebase.com/wiki/Data_dumps. Those files contain one assertion per line, and are made

to be easily converted into RDF or XML datasets [62]. Finally, the data of the collections of entities can be downloaded in RDF, TSV and CSV formats.

The number of points awarded by this dataset regarding the ease of accessing its data is summarised in the following table.

Concept	Comment	Points
SPARQL endpoint	No (MQL editor instead)	0
Number of results limited	n/a	0
Documentation	Yes	1
Dump files with RDF data	Yes	1
Other ways to retrieve data	Yes (CSV and TSV files)	1
TOTAL		3

Table 15: Freebase data accessibility

Number of overlaps

The animal collection of this dataset contains only 235 entities. “This is a collection of common-sense animals, more understandable to children than biologists” [63], which includes both general animals, such as bear (which represent a family of mammals) and bird (which refers to a whole class of animals) and more specific organisms such as grey wolf (*Canis lupus* species) and cheetah (*Acinonyx jubatus* species). The next figure shows the number of animal entities contained in this dataset.

Freebase Find topics... Data Schema Apps Docs

Animal

table started by [superrobert](#) for the [Biology Commons](#)
 There is no user-contributed description yet.
[Edit Description](#)

235 [Animal](#) topics [Filter this Collection](#)

Summary **Table** Gallery Map Timeline



Name	Image	Article
Alligator		Craniata An alligator is a crocodilian in the genus <i>Alligator</i> of th and the Chinese alligator (<i>Alligator sinensis</i>). The name alliga
Ant		Ants are social insects of the family Formicidae (/fɔːrˈmɪsɪdi/) s ancestors in the mid-Cretaceous period between 110 and 130

Figure 11: Freebase number of animal entities
 (<http://www.freebase.com/view/biology/views/animal>)

Interestingly, if a common or scientific name of a species not listed in the animal collection is searched in the search box of this site, probably it will be found. Any entity not present in the Freebase dataset will be searched in Wikipedia, and a snippet of the full text and an image from the same resource will be provided to the user. However, if the URI of the found entity is typed in a RDF browser, the displayed data contains only links and other metadata with no information about the species.

In relation to fish, this data source contains only two entities: fish and shark, which evidently are not suitable to be linked to any entity in the FishDelish dataset, as they do not correspond to specific fish species. Therefore, it can be concluded that the number of overlapping entities of this dataset is zero. Consequently, this resource is not considered as a prospect dataset for the construction of links.

3.3.11. OpenCyc

OpenCyc is the open source version of the Cyc system, developed by Cycorp, which is composed of a reasoning engine and a general knowledge base captured in an ontology in an OWL file. According to information on its site [64], the OpenCyc knowledge base is the largest and most comprehensive

in the world, and the latest version of this system contains hundreds of thousands of terms and millions of assertions. A more accurate number relating to the dimension of this ontology is not provided on the website. However, the execution of the application reveals the exact dimension of this dataset: 1,308,404 assertions and 417,671 terms, as evidenced in the following figure.

```

C:\Windows\system32\cmd.exe

;;; Load of KB 5018 completed <0:53> at 08/01/2011 15:30:01
;;; KB 5018 statistics
FORTs : 153968
Constants : 123193
cached indexing : 2 <0.00162%>
NARTs : 30775
cached indexing : 0 <0%>
cached HL formulas : 0 <0%>
Assertions : 1471304
KB Assertions : 1308404
cached : 0 <0%>
Bookkeeping Assertions : 162900
Deductions : 293378
cached : 0 <0%>
KB HL supports : 54872
cached : 0 <0%>
Unrepresented terms : 417671
cached indexing : 0 <0%>
Initialization time = 55.729 secs.

Start time: Mon Aug 01 15:30:02 BST 2011
Lisp implementation: Cycorp Java SubL Runtime Environment
JVM: Sun Microsystems Inc. Java HotSpot(TM) 64-Bit Server VM 1.6.0_17 <14.3-b01>

```

Figure 12: OpenCyc application execution

Both the open and commercial versions of this product share the same knowledge base. The difference between those two versions lies in extra features offered in the commercial one, including the complete version of the reasoning engine and other software tools.

Zaino [65] defines this knowledge base as an encyclopedia of general knowledge in machine-processable format, and points out that the development of this system began 25 years ago to address the deficiencies of expert systems regarding the discovery of further knowledge. They were only capable of reaching a solution to a problem without the ability to extend the knowledge of the domain.

No details or more information about the composition of the knowledge base are given in the website to indicate the topics that it covers, the structure of the ontology or the type of terms and assertions it contains.

Accessibility to the data

The OpenCyc system including the reasoning engine and the knowledge base can be downloaded in an executable file. Once the application is installed, it is the most effective and efficient way to find and explore the OpenCyc knowledge base, by using the different features to find, filter and explore

information. Furthermore, two other mechanisms to obtain data are available in the OpenCyc website. Firstly, there is a keyword search textbox that returns the URI and data of the found entity. Secondly, the full OpenCyc knowledge base can be downloaded in a single OWL file. No other ways to access data were found in the OpenCyc site or on the Web, such as a SPARQL endpoint or files in other formats (CSV, Excel, etc.).

The amount and quality of the documentation is good, as installation instructions, API reference guides, exercises and tutorials are supplied on the website. Thus, the score that this system receives regarding the accessibility to its information is presented below.

Concept	Comment	Points
SPARQL endpoint	No	0
Number of results limited	n/a	0
Documentation	Yes	1
Dump files with RDF data	Yes (the ontology file)	1
Other ways to retrieve data	Yes (OpenCyc App and keyword searches)	1
TOTAL		3

Table 16: OpenCyc data accessibility

Number of overlaps

In order to gauge the number of fish species and the type of information contained in this dataset, it was necessary to install the OpenCyc application.

In examining the application it was found that it comprises a Java web server that executes a web application that extracts and processes data from a set of files containing the knowledge base. The web-based interface is similar to the Protege ontology editor interface, where the user can explore in a hierarchical structure the list of classes and individuals of the ontology and the set of assertions about them.

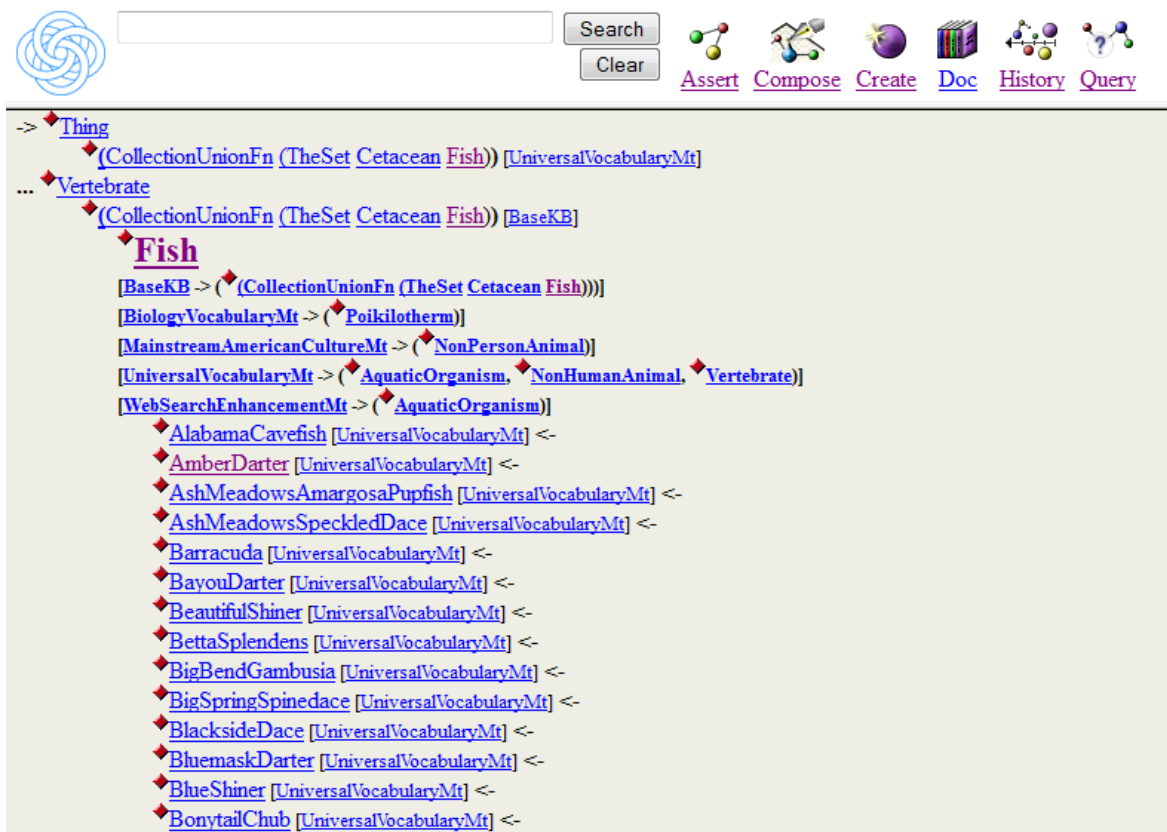


Figure 13: OpenCyc application interface

After studying the documentation, the collection of fish species was found. It contains 143 ontological classes of fish. However, this collection of fish is based primarily on common names of species and families, such as barracuda, carp, piranha and manta ray .Only one element in the list is named by its scientific name, the *Beta Splendens*. Some elements contained in this fish collection are not even fish, for example, the seahorse.

Thus, it can be concluded that this collection does not provide a biological or taxonomic structure of fish but a list of popular names of fish. This information can be helpful for other types of applications, but not for our purposes as this collection does not contain the scientific names of the vast majority of the elements, which are required to construct the links to FishDelish. Moreover, this collection is very heterogeneous in terms of taxonomy, mixing names of species with names of families and even with more generic groups such as “tropical fish” and “ray-fish”. Hence, the number of overlaps of this dataset is zero in practical terms.

Therefore, this dataset is not suitable for the generation of fish species links to connect it with the FishDelish dataset. On the other hand it is worth noting that this knowledge base indeed is comprehensive from an ontological perspective, as a large number of properties and assertions about the classes and individuals are contained in the ontology. For example, some assertions about

fish in the ontology include the following: fish are non-human animals; are bilaterally symmetric objects; are things; are vertebrates; and are natural resources.

3.3.12. Bio2RDF

Bio2RDF was created with the purpose of connecting different biological databases available on the Web with the use of semantic technologies to allow the generation of queries to search data in all the databases without the need of creating a central repository, according to Belleau et al. [66], who also describe Bio2RDF as a semantic mashup application that links the most popular bioinformatic databases.

Ansell et al. [67] list 31 databases converted to RDF format, being the following ones with the bigger number of triples:

- PubMed. A database of citations for biomedical literature from MEDLINE and other resources including life science journals and online books [68], with 797,000,000 triples.
- Uniparc. Database of protein sequences. Part of Uniprot, with 490,000,000 triples.
- Uniprot. Contains 338,602,962 triples.
- Uniref. “Provides clustered sets of sequences from UniProt Knowledgebase”[55] 242,000,000 with triples.
- NCBI GeneID. A database containing data about genes, with 172,931,628 triples.

The important database for our purposes is the Taxonomic database, which contains 3,877,201 triples.

The main components behind this project are a set of applications that transform data structured in a certain scheme into its equivalent RDF representation. These are known as “rdfizers”, and they use Xpath, regular expressions, SQL queries and other techniques to extract data from a wide range of resources including text files, relational databases, HTML and XML documents. [66]. Evidently, apart from the rdfizers, more tasks and components are required to perform the conversion to RDF, such as the normalisation of identifiers, the selection or creation of adequate namespaces and predicates and the installation and configuration of the underlying software to support the bio2rdf architecture, including application servers and triple stores.

Several SPARQL endpoints are available for this dataset. However, a SPARQL query must be executed in the right endpoint, otherwise, no results will be returned. At first this behaviour seems

odd, but the explanation for it is that this dataset has a distributed architecture with a triple store for each namespace. As Ansell et al. [67] explain, each triple store has its own domain name so the SPARQL endpoint can be accessed separately. As an example of this, the SPARQL endpoint for retrieving data from the PubMed database is [Http://pubmed.bio2rdf.org/sparql](http://pubmed.bio2rdf.org/sparql), while the endpoint related with the Uniprot database is <http://uniprot.bio2rdf.org/sparql>.

Accessibility to the data

The main method of finding data is the set of SPARQL endpoints, which are restricted to return no more than 10,000 triples.

Additionally, the Bio2RDF system offers the potential to perform a search of keywords in either a specific database or the whole dataset. The way to conduct a search is based on an HTTP API, which receives URL requests containing the parameters of the searches and returns whether an RDF or an HTML document depending on the client type. The documentation offers as an example the term propanol, which can be searched in the ChEBI database by requesting the URL <http://bio2rdf.org/searchns/chebi/propanolol>, while a search of the same word in all the databases can be done through the URL <http://bio2rdf.org/search/propanolol>

Finally, dump files in RDF format containing the different databases are available at <http://download.bio2rdf.org/data/>. Additionally, the Bio2RDF application can be downloaded for the creation of a local environment and consists of three components: a web application running on a Tomcat web server; the Virtuoso Open source community edition software to set up the SPARQL endpoint; and the collection of RDF files containing the databases.

The amount and quality of the technical documentation about this dataset is insufficient, and is focused on the installation of the application for the creation of mirrors, but not much information is provided about its use from a user's perspective. Even when some example SPARQL queries can be found on the Web, they are not intended to query the Taxonomy database, which is of interest to us. The score of this dataset in terms of data accessibility is presented below.

Concept	Comment	Points
SPARQL endpoint	Yes	1
Number of results limited	10000	0.5
Documentation	Yes, but insufficient	0.5
Dump files with RDF data	Yes	1
Other ways to retrieve data	Yes	1
TOTAL		4

Table : Bio2RDF data accessibility

Number of overlaps

In order to determine the number of overlapping species with the FishDelish dataset, the following query was executed to retrieve all the entities in the dataset that have a scientific name. It was not possible to construct a SPARQL query to retrieve only the fish species as the taxonomic classification used in this resource is not offered to the users, and any attempt to filter the data to get only the fish species may represent the loss of entities when the taxonomic system is not explicitly stated.

```
SELECT ?species ?scientific_name  
WHERE  
{  
?species <http://bio2rdf.org/ns/taxonomy#scientificName> ?scientific_name  
}
```

After retrieving some results, it was evident that the identifiers used in the URIs of the entities were similar to the ones used in the Uniprot dataset. A closer inspection revealed that the identifiers used in both datasets to refer to the same species are equal. Thus, given the URI of one entity in the Uniprot dataset, its corresponding entity in the Bio2RDF dataset can be known. For example, the URI of the species *Magnisudis atlantica* in Uniprot is <http://www.uniprot.org/taxonomy/319769>, while the URI for this species in Bio2RDF is <http://bio2rdf.org/taxonomy:319769>.

Some information in the CKAN catalogue [51] and in other resources on the Web confirms that indeed the Bio2RDF Taxonomy dataset was constructed with data from the Uniprot Taxonomy database. In order to reassure that both datasets contain the same species, a number of entities from Uniprot were randomly selected to verify their presence in the Bio2RDF dataset. The same procedure was conducted in reverse, that is to say that some entities in Bio2RDF Taxonomy were selected to confirm that they exist in the Uniprot dataset. As no element was found to be contained in one dataset but not the other, statistically it can be assumed that the vast majority of the Bio2RDF dataset overlaps the Uniprot taxonomy dataset.

Consequently, considering the information on the Web stating that the Bio2RDF data comes from Uniprot, the fact that the Bio2RDF project does not generate data but only takes existing data from other databases to convert it into RDF format, and the statistical verification using random samples, it can be assumed that the number of overlapping species between the FishDelish and Bio2RDF datasets is the same number of overlaps between FishDelish and Uniprot, possibly with a small

variation that does not represent an issue for the development of the project.

Evidently, the application of the same procedure that has been applied in other datasets to achieve the number of common species with the FishDelish dataset would have provided an accurate number of the overlaps of the Bio2RDF dataset. However, two factors hindered the application of the same procedure. Firstly, the restriction of only 2,000 results in the SPARQL endpoint made it difficult to retrieve the list of elements in this dataset. Filtering the dataset by retrieving only the fish species has been done in situations where the dataset is too large. However, as the taxonomic classification used in this dataset cannot be clearly known, the decision was made not to filter this list to avoid the loss of entities.

The second factor was the large size of the RDF dump file. The next attempt to get the list of species in this dataset consisted of downloading the dump file to perform queries in a local environment, which is based on an OpenRDF Sesame server, in order to avoid the limitation of the number of results. However, due to the large size of that file (almost 0.5 GB) it was not possible to load it into the server.

Number of assertions per species

The score of Bio2RDF for this factor is provided in the following table.

Species	URI	No. of assertions
<i>Centropyge flavicauda</i> (Whitetail angelfish)	http://bio2rdf.org/taxonomy:109727	13
<i>Coryphaenoides acrolepis</i> (Pacific grenadier)	http://bio2rdf.org/taxonomy:83390	9
<i>Labeobarbus aeneus</i> (smallmouth yellowfish)	http://bio2rdf.org/taxonomy:137081	10
<i>Merluccius gayi gayi</i>	http://bio2rdf.org/taxonomy:307687	6
<i>Pseudopleuronectes yokohamae</i> (marbled flounder)	http://bio2rdf.org/taxonomy:245875	12
<i>Sillago robusta</i> (stout stillago)	http://bio2rdf.org/taxonomy:443725	9
<i>Takifugu pseudommus</i> (nameradamashi)	http://bio2rdf.org/taxonomy:96884	12
<i>Zosterisessor ophiocephalus</i> (Gobius ophiocephalus)	http://bio2rdf.org/taxonomy:85428	15
<i>Gymnothorax flavimarginatus</i> (yellow-edged moray)	http://bio2rdf.org/taxonomy:217850	15
<i>Odaxothrissa vittata</i> (Regan's fangtooth pellonuline)	http://bio2rdf.org/taxonomy:402404	9
Average	11	

Table 17: Bio2RDF estimated number of triples per species

3.3.13. Global Names Index

The Global Names Architecture (GNA) is described on its website [69] as a collection of databases, applications and services to find, organise and interconnect data about species, focusing on their names. It is supported by the US National Science Foundation and was started 10 years ago with the aim of compiling information about names of organisms available in the Web in order to support taxonomists, biodiversity informaticians and nomenclaturalists in doing their jobs more efficiently.

One of the main components of GNA is the Global Names Index (GNI), which is a comprehensive list of names in binominal nomenclature developed by the Global Biodiversity Information Facility (GBIF) and the Encyclopedia of Life. It contains 19,384,364 names (July 2011), which is a large number compared to the number of species provided in the GNA website (1,900,000). This difference between the number of names and species, according to information on the same website, gives an idea of the dimension of the “many names for one species” problem. However, it also states that most of the causes of this problem are “alternative spellings of the same name”, including misspellings, truncations and other similar errors. The other big problem in the field of species’ nomenclature to be addressed is the problem of “one name for many species”.

Accessibility to the data

Following comprehensive research on the Web about the mechanisms to access data from this dataset, only two ways were found. The first is a standard search textbox on the website, which, when given a name, returns a set of records matching the input including the names of the datasets that contain those records.

The second mechanism is an API based on queries constructed with URLs that, upon request, return an XML document with data of the found occurrences as a response. However, the functionality of this API is very limited as it is not possible to retrieve groups of species contained in a taxonomic clade, or to retrieve the collection of all the names in the dataset. This API is useful for the identification of records of a particular species, but not for the extraction of groups of species.

Another characteristic of this dataset that makes it difficult the generation of RDF links is that, given an input string, multiple records are returned that correspond to the same species, but with minor variations. As a result, it is not possible to establish unique mapping between the elements from the base dataset and these of GNI. As an example of this problem, let us consider the case of the name *Pangasius bocourti*. A query made with the API returns the following list of names:

Pangasius (Pseudopangasius) bocourti Sauv.
Pangasius (Pseudopangasius) bocourti Sauvage 1880
Pangasius (Pseudopangasius) bocourti Sauvage 1880 sec. Eschmeyer 2004
Pangasius bocourti
Pangasius bocourti
Pangasius bocourti Sauvage 1880
Pangasius bocourti Sauvage, 1880

The search of more popular species may return a much bigger number of occurrences, which is the case of *Salmo salar*, that returns 125 matches.

In conclusion, both technical and conceptual issues with this dataset make the generation of appropriate links difficult, given the lack of tools to extract automatically data of groups of species, and the multiplicity of entities that match a given name. For those reasons, this dataset is not suitable to be linked to FishDelish.

3.3.14. Decision matrix construction

A decision matrix is a suitable tool for the selection of the datasets that better satisfy the required elements for the construction of valuable links with our base dataset.

The general procedure for the construction of a decision matrix is described below:

1. Assign weights. Some factors are more significant than others. Therefore, assigning weights to the factors allows direct comparison [70].
2. Normalise the factor values to a standardised scale. Usually, the values of the factors are arbitrary numbers. Thus, a normalisation of those numbers is required to perform a correct comparison.
3. Threshold selection. The maximum possible score that an alternative can attain is calculated by multiplying each weight by the maximum possible value that a factor can get. Then, a percentage of that number must be selected and calculated, which will represent the minimum value that an alternative must get in order to be selected.
4. Calculate total scores for each alternative. The value of each factor will be multiplied by the corresponding weight.
5. Identify the alternatives that scored higher than the selected threshold. Those alternatives

are selected.

<i>Factor</i>	<i>Weight</i>	<i>Alternative₁</i>	<i>Alternative₂</i>	<i>Alternative₃</i>	<i>Alternative_n</i>
Factor 1	W_1	D_{11}	D_{21}	D_{31}	D_{n1}
Factor 2	W_2	D_{12}	D_{22}	D_{32}	D_{n2}
Factor 3	W_3	D_{13}	D_{23}	D_{33}	D_{n3}
Factor p	W_p	D_{1p}	D_{2p}	D_{3p}	D_{np}
Total score		T_1	T_2	T_3	T_n

$$T_i = W_1 D_{i1} + W_2 D_{i2} + W_3 D_{i3} + \dots + W_p D_{ip}$$

Figure 14: General decision matrix

Weight assignment

The estimated number of triples per species is an indicator of the amount of information that each dataset offers about fish species. A dataset that offers more assertions is likely to contain more useful information that the users can be interested in. Moreover, the application that was developed to demonstrate the use of RDF links can extract more data from the datasets containing more triples.

The accessibility to the data factor provides an estimation of the technological infrastructure that supports the resource of information. A site that offers more mechanisms for finding and retrieving its data is a better site from the technical perspective. Additionally, this factor indicates how easy or difficult the extraction of the required information is, as the construction of links requires the retrieval of large amounts of data from the different datasets. Therefore, a data source whose data is more technically accessible, is more suitable for the construction of links. Additionally, the development of the example application depends on the possibility to extract data from the selected datasets.

The number of overlaps is also important. A dataset with no overlapping species is useless for the construction of links. Conversely, a dataset that contains a large number of fish species' entities that are contained in the base dataset, allows the generation of more links. Thus, more overlaps represents more links.

Therefore, based on the previous arguments, it was concluded that the three factors have the same importance. As a result, the same weight was assigned to every factor. To simplify the mathematical process, the selected weight was 1.

Normalisation of values

In order to normalise the values of the different factors, a 0-5 scale was used. For each factor, the dataset with the highest score was awarded 5 points, and the remaining values are calculated accordingly. In the decision matrix, both the raw and normalised scores are provided for each factor. The first column of each factor contains the raw score, while the next column shows the normalised value.

Threshold selection

Once the total scores for every dataset were attained, the average score was calculated, and the datasets above this value were selected.

Calculation of the total score for each dataset

Factor	Accessibility		Number of overlaps		Number of triples		Total Score
<i>Weight</i>	1		1		1		
BBC Wildlife Finder	2.5	2.5	39	0.018	105	4.2	6.72
Uniprot	4.5	4.5	10541	5	12.2	0.488	9.99
EUNIS	3	3	2250	1.06	108.4	4.336	8.4
Geospecies	4	4	28	0.013	107.9	4.316	8.33
Taxonconcept	4	4	6844	3.25	125	5	12.25
DBpedia	4	4	8151	3.87	37	1.48	9.35
Bio2RDF	4	4	10541	5	11	0.44	9.44

Table 18: Decision Matrix

The decision matrix does not contain the datasets that were previously rejected for different reasons.

3.3.15. Analysis of results

The average score was 9.21. Therefore, the selected datasets were those having a score higher than this value: Uniprot, Taxonconcept, Dbpedia and Bio2RDF. On the other hand, three datasets were rejected using the decision matrix: BBC Wildlife Finder, Geospecies and EUNIS. In particular, both BBC Wildlife Finder and Geospecies contain a reduced number of overlaps (39 and 28 respectively), which are insignificant numbers when considering the size of FishDelish, which contains almost 32,000 entities. EUNIS provides 2250 overlaps, which represents less than one-third of the number of overlaps that offers the selected dataset with the lowest number of overlaps (Taxonconcept).

Finally, five other datasets were not considered in this evaluation for three main reasons:

- Its data was not accessible due to the lack of mechanisms to find and retrieve it.
- The dataset did not contain any entity that can be linked to the fish species in FishDelish.
- It was not possible to identify uniquely an entity corresponding to a scientific name.

Fishes of Texas and Spire were rejected for the first reason, while Freebase and OpenCyc were discarded based on the second reason. A combination of the first and third reasons was the cause of rejecting Global Names Index.

In summary, of the 12 datasets analysed, 5 were not evaluated, 3 were evaluated but not selected, and 4 were selected.

3.4. Identification and Selection of Predicates

The next step for the construction of the RDF links was the identification and selection of the most appropriate terms that satisfied the expressivity and popularity requirements. Specifically, a vocabulary was required containing a set of terms to link URI aliases expressing different levels of equivalence.

The links that were generated correspond to links connecting URI aliases of fish species between the base and target datasets. In this respect, a discussion has been initiated in the background section about the most appropriate vocabularies and predicates for this purpose. This concluded that even when the most common term used in this scenario is *owl:sameAs*, predicates from other vocabularies may provide a more appropriate connection between the involved datasets.

As Bechhofer and Miles [15] state, *owl:sameAs* is usually inappropriate to link entities from different datasets, because the resulting formal consequences may not be desired. The same reasoning applies to other similar OWL terms: *owl:equivalentClass* and *owl:equivalentProperty*.

Another possible solution was the creation of *ad hoc* predicates that satisfied completely our requirements in terms of semantic. Nevertheless, the development of new terms is recommended only in situations where existing vocabularies are not capable of describing the particular relationship. Heath and Bizer [2] suggest the reuse of existing terms wherever possible, as some applications may not be able to process unknown vocabularies.

A predicate was required that could indicate that two entities refer to the same object of the real world (URI aliases) but without logical implications, to avoid logical inconsistencies in applications and reasoning engines. Additionally, it was needed to express different types of matches: one type for fish species entities that have the same scientific name; and another type for entities that not only have the same scientific name but also the same authority.

Even when it is sure that two entities refer to the same species if they have the same scientific name, the authority adds an extra level of equivalence. Frequently the scientific names in the scientific literature are followed by the authority, which, according Ruppert and Fox [71], consists of the name of the author or authors who named the species and described it by first time as well as the date of the description. Thus, it was convenient to express that the URI aliases with the same authority have a tighter relationship than those that match only in the scientific name.

3.4.1. Identification of Vocabularies and Predicates

The identification of potential vocabularies has been conducted with the use of catalogues and other related resources. There is no a comprehensive catalogue containing all the available vocabularies. However, those catalogues generally contain the most popular vocabularies used in the Web of Data. Therefore, as popularity is a key requirement, there is no interest in finding further vocabularies not presented in the catalogues with other mechanisms.

Most vocabularies are constructed to describe aspects of very specific topics, such as music, bibliographic references, food or geographical locations, while the number of more general vocabularies that provide terms that can be used in different subjects is reduced. This fact facilitated the identification of prospect vocabularies in catalogues, as most were rejected because they contained terms for other domains of interest.

Three catalogues were used: SchemaWeb [72], LOV (Linked Open Vocabularies) [73] and SchemaCache [74], which are some of the most popular and visible vocabulary catalogues on the Web. As on July 2011, SchemaWeb contains a list of 241 vocabularies, LOV holds 130 and SchemaCache contains 360.

The following vocabularies were found that provide some terms that may be used for the construction of URI aliases.

3.4.1.1. *MuSim, the Similarity Ontology*

This is an ontology designed to express similarity between two or more things. Initially, it was created to express different types of similarity in the music domain, but it is thought to be applied in a variety of domains. The interesting characteristic of this ontology is that the association between two entities is stated by the use of concepts instead of properties, which can be reified to specify the type of association [75].

Jacobson [76] indicates that similarity is a broad concept and usually two things are similar in a specific sense. This approach of using classes instead of properties allows to express in what sense two things are similar.

There is a class to describe similarity (*sim:Similarity*) and another to describe the method for determining the similarity (*sim:AssociationMethod*). The use of these two classes in conjunction allows the expression of how the associated entities are similar. Therefore, this vocabulary fulfils our needs in relation to expressivity, as the different levels of relationship between fish species entities can be conveyed with the use of those two concepts.

3.4.1.2. *Biological Taxonomy Vocabulary*

The Biological Taxonomy Vocabulary [77] offers a set of terms to specify the taxonomic classification of any kind of organism, and also provides other specialised terms for botany and zoology. Some of the terms in this vocabulary are *biol:genus*, *biol:species*, *biol:authority*, *biol:kingdom* and *biol:commonName*, which can be very useful in the bioinformatics domain.

There is one property in this vocabulary that is thought to link two elements in different datasets referring to the same form of life: *biol:seeAlso*. This predicate would have been perfect for our needs if only one type of relationship was required to create the links. But a vocabulary with a collection of terms to express different levels of equivalence is required, and this dataset only offers one term for this purpose. Thus, this vocabulary does not meet our expressivity needs.

3.4.1.3. *Taxonconcept Ontology*

This ontology was created to structure the data in the Taxonconcept dataset. It provides some specific terms to create links with other popular data sources, for instance, *txn:hasWikipediaArticle*, *txn:hasEUNISPage* and *txn:hasITISPage*. However, those terms are intended for the creation of links to non-Linked Data resources, such as Wikipedia or ITIS. The predicate used in Taxonconcept to link species with other RDF data sources is *skos:closeMatch*. Hence, this vocabulary is not suitable for

our purposes.

3.4.1.4. The Association Ontology

This ontology offers basic concepts and properties for describing association statements [78]. However, this vocabulary is intended mainly to provide terms to associate things to people, but those terms are very abstract that may be applied in other domains. The main term of this ontology is *sim:Association* and this can be used to express that two entities are related or associated. Nevertheless, this term is very vague and abstract to indicate that two entities refer to the same species. Moreover, this vocabulary does not provide more terms that could be used to link URI aliases with different levels of equivalence.

3.4.1.5. SKOS

An overview of this vocabulary is provided in the background section of this study. A set of properties to state mapping links between concepts in organisation systems is contained in this vocabulary. Two types of mapping properties are provided, hierarchical and associative. “A hierarchical link between two concepts indicates that one is in some way more general ('broader') than the other ('narrower'). An associative link between two concepts indicates that the two are inherently 'related', but that one is not in any way more general than the other” [15]. The properties used to state hierarchical mapping links to connect two entities are *skos:narrowMatch* and *skos:broadMatch*, while the associative properties are *skos:relatedMatch*, *skos:closeMatch* and *skos:exactMatch*.

skos:relatedMatch is used to express associative mapping links. *skos:closeMatch* can link two similar entities that can be used interchangeably in some applications, while *skos:exactMatch* is used to link entities while expressing a “high degree of confidence that the concepts can be used interchangeably across a wide range of information retrieval applications” [15].

The linking properties in SKOS satisfy our expressivity requirements as they allow different degrees of association of URI aliases.

3.4.2. Selection of the Predicates

Heath and Bizer [2] suggest consideration of the following factors for the selection of vocabularies: the popularity, maintainability and expressivity of the vocabulary.

The level of maintenance of a vocabulary can be difficult to measure, as it is dependent on the availability of information in the sites about this factor. Therefore, the two characteristics considered were the popularity and the expressivity of the vocabulary.

To estimate the popularity of the predicates from the different vocabularies that were found, a SPARQL query was executed in the LOD cloud to calculate the number of triples that contain each predicate. This procedure was used in other project referred to in the background section of this paper to assess the popularity of RDF terms. The SPARQL endpoint that was used is <http://lod.openlinksw.com/sparql>, which contains a cache of the LOD cloud.

Vocabulary	Term	No. of occurrences	Query
The Similarity Ontology	<i>sim:Similarity</i>	0	SELECT count(*) WHERE {?s <http://purl.org/ontology/similarity/Similarity> ?o}
	<i>sim:AssociationMethod</i>	0	SELECT count(*) WHERE {?s <http://purl.org/ontology/similarity/AssociationMethod> ?o}
Biological Taxonomy Vocabulary	<i>biol:seeAlso</i>	5	SELECT count(*) WHERE {?s <http://purl.org/NET/biol/ns#seeAlso> ?o}
The Association Ontology	<i>sim:Association</i>	0	SELECT count(*) WHERE {?s <http://purl.org/ontology/similarity/Association> ?o}
SKOS	<i>skos:exactMatch</i>	1579024	SELECT count(*) WHERE {?s <http://www.w3.org/2004/02/skos/core#exactMatch> ?o}
	<i>skos:closeMatch</i>	453773	SELECT count(*) WHERE {?s <http://www.w3.org/2004/02/skos/core#closeMatch> ?o}

Table 19: Estimation of the popularity of prospective predicates

The table above demonstrates the number of occurrences of each predicate in the LOD cloud cache.

Although some vocabularies from the biological domain, and others for general purposes, were identified containing terms that may be used for the generation of links connecting URI aliases, most lack the expressivity required to define different types of matches. Hence, only two vocabularies met our expressivity requirements: SKOS and The Similarity Ontology. However, there are no triples in

the LOD cloud cache containing the predicates of the Similarity Ontology, while the number of triples that contain the SKOS terms is very large. Undoubtedly the most popular of the found terms are the SKOS predicates.

Consequently, it can be concluded that the only vocabulary that offers the required popularity and expressivity is SKOS. Furthermore, SKOS is a W3C recommendation, and its predicates have been used by other species data providers (Taxonconcept and Geospecies) to establish links to other datasets.

The term *skos:exactMatch* was used to link entities that match in both their scientific names and authority, while the predicate *skos:closeMatch* was selected to link species that match only in their scientific name, based on the description of these predicates.

It is noteworthy that the triple structure has some limitations in providing more details about the relationship between the subject and object in an assertion, as the predicate only contains a piece of data. Some approaches, such as that used in the Similarity Ontology, offer a mechanism to solve this problem, with the use of an intermediate object between the two linked entities. The following figure shows the graph of a standard RDF link and an example of a link created with an intermediate element.

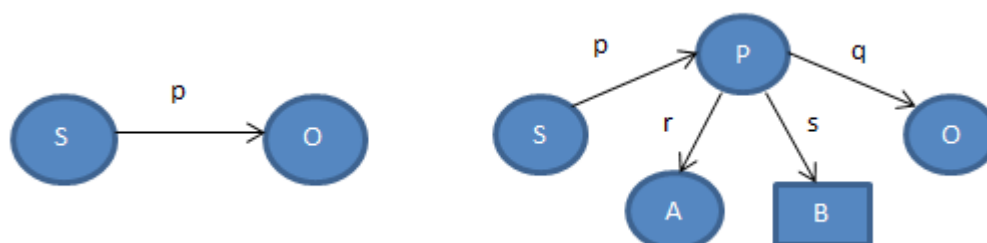


Figure 15: Standard and extended link graph

This alternative linking of entities offers the possibility of providing further metadata about the relationship of the two elements, which has some applications such as the inclusion of metadata about the link (i.e., when it was created, who created it) and the description of the provenance of data items.

In our case, the simple triple structure was sufficient for the creation of links because it was not necessary the inclusion of more metadata to describe the relationship between the fish species entities.

3.5. Construction of the RDF Triples

Once the predicates were selected, the next step was to assign the right predicate to link each of the URI aliases. The authority was taken into account in selecting the term joining each pair of entities. However, the comparison of the authority represented new challenges as a literal comparison of strings was not useful always. Thus, a set of operations were applied to the authority strings in order to compared them, and in the cases where the exact string matching did not work, an approximate string matching technique was used to compare them and select the right term to create the link.

As discussed previously, the Dataset Comparison Application allows the identification of URI aliases by performing a literal string comparison of the scientific names of the species. Additionally, two predicates have been selected to construct the links: *skos:exactMatch* and *skos:closeMatch*. The next challenge lay in identifying the cases where one or other predicate had to be used.

In order to decide which predicate should be used for each link, the authority of the fish species was compared. A literal string comparison was not always useful in this case, because some of the names can vary from one dataset to other. Differences in the spelling, use of abbreviations, names in other languages, use of diacritical marks and other characters from other languages, are some of the differences that can be noticed while comparing names of authors from different datasets. For instance, the authority of the species *Serrasalmus geryi* in FishDelish is “Jegu and Dos Santos, 1988”, and in Uniprot it is “Jegu & Santos, 1988”. Evidently those two strings refer to the same authority, but a literal string comparison would fail to recognise them as the same authority. Hence, a suitable method to compare strings was applied in those cases.

The first task involved extracting the authority data from the different datasets. The data that initially was retrieved from the analysed datasets only contained the URIs and the scientific names of the species.

The following queries were executed to retrieve the authority in addition to the URIs and scientific names:

FishDelish:

```
PREFIX fd: <http://fishdelish.cs.man.ac.uk/rdf/vocab/resource/>
SELECT ?element ?species ?genus ?author
WHERE {
    {
        ?element fd:species_Species ?species .
        ?element fd:species_Genus ?genus .
        ?element fd:species_Author ?author .
    }
}
```

DBpedia:

```
PREFIX dbpprop: <http://dbpedia.org/property/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
SELECT ?element, ?binominal, ?author WHERE {
    ?element dbpprop:binomial ?binominal .
    OPTIONAL {?element dbpprop:binomialAuthority ?author} .
    {?element dbpedia-owl:class dbpedia:Actinopterygii}
    UNION
    {?element dbpedia-owl:class dbpedia:Cephalaspidomorphi}
    UNION
    {?element dbpedia-owl:class dbpedia:Chondrichthyes}
    UNION
    {?element dbpedia-owl:class dbpedia:Myxini}
    UNION
    {?element dbpedia-owl:class dbpedia:Petromizontida}
    UNION
    {?element dbpedia-owl:class dbpedia:Placodermi}
    UNION
    {?element dbpedia-owl:class dbpedia:Sarcopterygii}
}
ORDER BY ?binominal
```

Taxonconcept:

```
PREFIX txn: <http://lod.taxonconcept.org/ontology/txn.owl#>  
  
SELECT ?species, ?scientific_name, ?authority  
WHERE  
{  
  ?species txn:hasScientificName ?scientific_name.  
  OPTIONAL {?species txn:scientificNameAuthorship ?authority}  
}
```

It is worth noting that the use of the keyword OPTIONAL was fundamental in the last two queries because the authority triple is not provided in a considerable number of elements of the target datasets. Thus, the execution of those queries without that particular keyword would not retrieve the elements not containing this data.

The RDF file downloaded previously from Uniprot already contained the authority information. With regard to Bio2RDF, the links to this dataset have been created based on the generated links to the Uniprot dataset, because those two datasets contain the same species, according to a previous explanation.

3.5.1. Preprocessing of the Authority Strings

After analysing the authority strings from the different datasets it was noted that a literal string comparison could be used to identify the vast majority of the matches if string preprocessing is applied in the following ways:

1. Conversion of the strings to lower-case. A basic operation that usually must be applied for the comparison of strings.
2. Removal of diacritical marks. Some datasets contain names with diacritical marks used in other languages, while the author names in FishDelish do not contain those characters. For instance, “Fernández-Yépez, 1950”, “Lacépède, 1803” and “Forsskål, 1775” are strings contained in DBpedia, and “Jørgensen 1912” and “Küçük 2007” are contained in

Taxonconcept.

3. Removal of parentheses. A large number of authors are between parentheses in the different datasets. According to Ruppert and Fox [71], if an author's name is in parentheses it indicates that the scientific name has been changed since it was first named. However, the use of this mark is irregular. In one dataset the author of a species may be in parentheses and, simultaneously in another dataset, the author of the same species may not. Thus, the removal of this mark could increase the number of authority matches.
4. Conversion of "&," and "&" for "and". The authority of some species includes more than one author. However, even inside the same dataset the use of this conjunction is not homogenised. "Quoy and Gaimard, 1825" or "Stephens & Hobson, 1966" are examples of this issue.

After preprocessing the authority strings with the previous operations, the number of overlapping elements that matched in the authority increased significantly.

The following table demonstrates the number of overlaps with the same author before and after applying the normalisation functions, using literal string comparisons.

Dataset	Number of overlaps	Overlaps same authority before normalisation	Overlaps same authority after normalisation	Authority not provided	Number of cases where the literal string comparisons was not useful
Dbpedia	8151	3092	4599	3202	350
Uniprot	10541	6893	8013	2185	343
Taxonconcept	6844	2867	3285	3354	205

Table 20: Number of authority matches before and after the string preprocessing

An important fact that must be highlighted is that the authority does not match in a large number of cases because this data is not provided in the target datasets. Nevertheless, this data is present in all elements of FishDelish.

As can be seen in the previous table, most of the records that match in the authority have been identified already with the use of literal comparisons of preprocessed strings. However, there are still a number of cases where the literal comparison was not useful in deciding whether or not the authority was the same. Therefore, a suitable approximate string matching technique was applied.

3.5.2. Approximate Matching of the Authority Strings

In computer-based information systems, typing and spelling errors are a common cause of variation in strings, according to Hall and Dowling [79], who also report that the most common errors are the omission of a letter, the substitution of one letter for another and the insertion of an extra letter. They also state that another problematic source of variation of strings is abbreviations, particularly in the writing of names.

The concept of approximate string matching refers to the the process of comparing strings to understand how similar they are. The similarity of the strings is called the edit distance. The edit distance is a number resulting from the sum of the number of operations that must be applied to transform one string into the other [80].

The edit distance is calculated with a similarity function, which receives as input two strings s and t and produces an integer number $d(s,t)$. The similarity of two identical strings is 0 and this value increases if they are different. The problem of comparing similar strings can be solved by finding the pairs (s,t) such that $d(s,t)$ is greater than a given threshold of acceptability [79].

A more formal definition is provided by Navarro [80]: “The distance $d(s,t)$ between two strings s and t is the minimal cost of a sequence of operations that transform s into t (and ∞ if no such sequence exists). The cost of a sequence of operations is the sum of the costs of the individual operations”.

Navarro [80] comments that the edit distance concept is so general and powerful that can be applied in a wide range of applications, as most of the algorithms created to calculate this value can be modified easily to solve specialised and complex problems. The main application areas of this technique are computational biology, signal processing and text retrieval.

In computational biology, the edit distance is used to compare and search for DNA and protein sequences, which are represented as strings over particular alphabets; for example, $\{A,G,C,T\}$. Literal string comparisons are not very useful in this field as the experimental measures usually contain errors that produce inaccurate strings representing the DNA and protein sequences, and sometimes the correct chains may contain some variations. Thus, computational biology has been developed with the use of this type of string comparison algorithms [80].

The most common functions to calculate the distance between two strings, according to Navarro, are:

- Levenshtein or edit distance. Three types of operations are allowed: insertions, deletions and replacements. All operations have the same cost (1).
- Hamming distance. Allows only substitutions with a cost of one each.
- Episode distance. Allows only insertions with a cost of one each.
- Longest Common Subsequence distance. Only deletions and insertions are permitted.

The selection of any of the previous algorithms depends on the specific characteristics of the problem. According to Navarro, the Hamming distance algorithm is useful to know the number of characters that mismatch between two strings. The Episode distance is used to track a sequence of events because it only allows insertions, and sometimes it is not possible to convert s into t with this function. The Longest Common Subsequence distance allows the measurement of the maximum possible length of a common sub-string of two strings. Finally, the Levenshtein distance is suitable for measuring the number of operations that must be performed to make two strings equal [80]. Therefore, the most adequate function for finding the similarity between two strings containing names of authors and years is the Levenshtein algorithm.

The next step entailed selecting an adequate threshold. To find the threshold, the edit distance values were calculated of the authority strings that did not match with the literal comparison. The Levenshtein distances were found with the use of the Dataset Comparison Application and an implementation of this function provided by the Apache Jakarta Commons project [81].

After analysing the results, it could be noticed that the Levenshtein distances measuring less than four correspond clearly to strings referring to the same authority. As an example, the following table contains some of the calculated values.

String dataset A	String dataset B	Levenshtein Distance
forsskal, 1775	fosskal, 1775	1
lewis, 1982	lewis, 1982	1
springer and d'aubrey, 1972	springer and daubrey, 1972	1
greenwood, 1973	greenwood, 1974	1
johnson and brothers, 1989	johnsons and brothers, 1989	1
kiener, 1963	keiner, 1963	2
collette, starck and phillips, 1974	collette, stark and phillips 1974	2
ranzani, 1839	ranzani, 1840	2
linnaeus, 1766	linneaus, 1766	2

goode and bean, 1882	good and bean, 1882!	2
garzon-ferreira and acero p., 1991	garzon-ferreira and acero, 1991	3
melendez and markle, 1997	melendez c. and markle, 1997	3
orces, 1960	orces v., 1960	3
reis and lehmann, 2009	reis and lehmann a., 2009	3
sousa and rapp py-daniel, 2005	de sousa and rapp py-daniel, 2005	3

Table 21: Examples of Levenshtein distances less than four

However, in most cases an edit distance larger than three does not reflect that the compared strings refer to different authority. One problem is that some authors have more than one name. In some records only one name is provided while, in others, second names are also given. In the following table some pairs of strings with a Levenshtein distance larger than three are provided as example.

String dataset A	String dataset B	Levenshtein Distance
contreras-balderas and lozano-vilano, 1994	contreras-balderas and lourdes lozano, 1994	9
ohridanus karaman, 1924	karaman, 1924	10
vaz-ferreira and sierra de soriano, 1974	vaz-ferreira and sierra, 1974	11
jordan and evermann, 1896	jordan and evermann in evermann, 1896	12
bocage and capello, 1864	barbosa du bocage and de brito capello, 1864	20
randall and burgess in burgess and axelrod 1972	randall and burgess 1972	23
lowe, 1841	philippi, 1887	9
gunther, 1864	eigenmann, 1909	10
richardson, 1844	eschmeyer, 1983	11
richardson, 1846	moreland, 1960	12
peters, 1866	ramsay and ogilby, 1886	18
bloch and schneider, 1801	walbaum, 1792	21

Table 22: Examples of Levenshtein distances larger than three

It can be noted from the previous table that the first six pairs refer to the same authority, while the next six pairs refer to different authors and years.

As an automatic approach or algorithm could hardly detect if a pair of strings with a Levenshtein distance larger than the selected threshold correspond to the same authority or not, a manual procedure was performed to solve this problem. In addition, the number of cases where the edit distance is larger than three is reduced. For instance, only 48 pairs out of the 205 cases in Taxonconcept where the literal comparison of strings was not useful, have a distance larger than

three. Thus, the user interaction was required in those few cases to compare the authority and select the right term for the construction of each link.

However, the use of a low threshold does not warranty total accuracy, because the existence of pairs of strings referring to different authority with an edit distance less than four is not impossible, even when it is unlikely. For that reason, a sample of 90 pairs of strings with a Levenshtein distance less than four was examined manually to confirm that the selected threshold was adequate. No pairs referring to different authority were found, but it might be possible that a very reduced number of links have been created with an incorrect predicate.

3.5.3. Algorithm Used for the Construction of the Links

A link is created between an element from FishDelish and an element from one of the target datasets if they have the same scientific name. The selection of the predicate to link two elements depends on the authority. If any of the elements does not contain the authority, *skos:closeMatch* is used. If the authority strings are exactly equal, the used predicate is *skos:exactMatch*. Otherwise, the Levenshtein distance of the two strings is calculated. If the value of the distance is lower than four, *skos:exactMatch* is used. If it is larger than three, the user decides if those strings refer to the same authority or not. If they refer to the same authority according to the user input, again *skos:exactMatch* is used. If not, *skos:closeMatch* is selected. The following table summarises which predicate is used in each situation.

Case	Predicate
Same scientific name and same authority	<i>skos:exactMatch</i>
Same scientific name and unknown authority in one or the two elements	<i>skos:closeMatch</i>
Same scientific name but different authority	<i>skos:closeMatch</i>

Table 23: Predicates used for the construction of links

The final algorithm for the generation of links is presented below.

```

For each entity a in Dataset A{
  For each entity b in Dataset B{
    if a.scientific name == b.scientific name{
      if(a.authority!= null and b.authority!=null){
        if(a.authority == b.authority){
          new link using skos:exactMatch
        }else{
          if(Levenshtein(a.authority,b.authority<4)){
            new link using skos:exactMatch
          }
        }
      }
    }
  }
}

```

```

    }else{
        if(user input == yes)
            new link using skos:exactMatch
        else
            new link using skos:closeMatch
    }
}
}
}

```

Once the overlapping elements were identified by comparing the scientific names and the correct predicate was selected for each case according to the authority, the generation of links in RDF files was performed easily with the application used during the development of this project. It is worth noting that the manual decision of the equivalence of some pairs of authority strings was feasible because the number of times that it was required the user interaction was reduced, as most pair of strings matched with the literal comparison or had a Levenshtein distance lower than four. Only 387 cases from all the datasets (173 from DBpedia, 48 from Taxonconcept and 166 from Uniprot) were decided manually.

3.5.4. Number of Links Created

The number of RDF triples created to link FishDelish with the four selected datasets is presented below.

	DBpedia	Uniprot	Bio2RDF	Taxonconcept	Total
No. of overlaps same authority	4872	8267	8267	3486	24892
No. of overlaps unknown authority	3202	2185	2185	3354	10926
No. of overlaps different authority	77	89	89	4	259
Total number of links generated per dataset	8151	10541	10541	6844	36077

Table 24: Number of RDF links generated

The number of RDF triples that have the *skos:exactMatch* predicate is 24,892 (69%), while the number of triples that contain the *skos:closeMatch* term is 11,185 (31%). As it was previously explained, the *skos:exactMatch* was used to link elements with the same scientific name and

authority, and the *skos:closeMatch* was used to link elements that only matched in the scientific name.

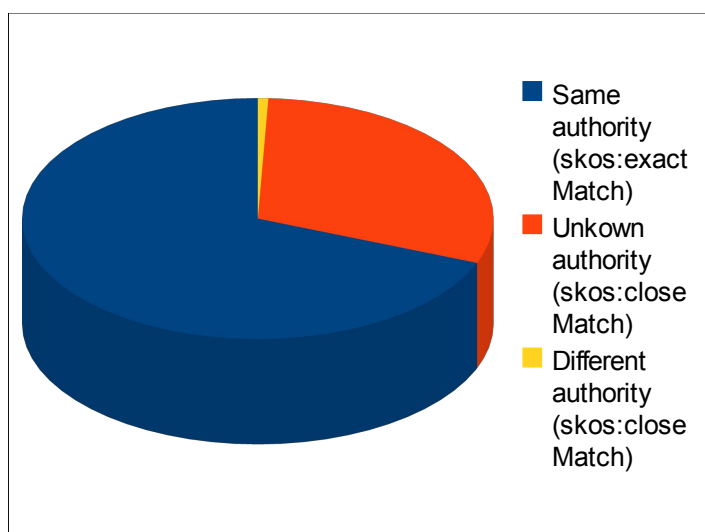


Figure 16: Proportion of links containing *skos:exactMatch* and *skos:closeMatch*

One conclusion that can be drawn from the previous graph is that most of the overlapping elements have the same authority if it is provided. However, in a large number of cases, the authority was not available. The authority was different in only a few cases, which indicates that, in general, the different data sources agree on the authority of the species.

The final number of links constructed to connect FishDelish with the four selected datasets is 36,077. The number of fish species in FishDelish that have at least one link to any of the other datasets is 15,404. Thus, 48.25% of the 31,927 elements in FishDelish were linked to other resources. This means that virtually one half of the fish species in the base dataset were linked to other sources of information.

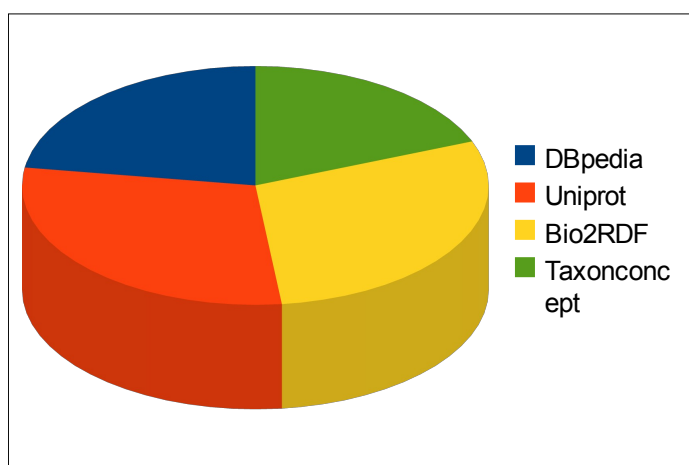


Figure 17: Proportion of links connecting FishDelish with the other datasets

An RDF file containing the triples that link the FishDelish entities was created for each dataset. As an example of the generated links, four triples that correspond to the links that connect the element *Aptosyax grypus* to the other four datasets are presented in N-Triple format:

```
<http://fishdelish.cs.man.ac.uk/rdf/species/Aptosyax/grypus>  
<http://www.w3.org/2004/02/skos/core#closeMatch>  
<http://bio2rdf.org/taxonomy:143610>.
```

```
<http://fishdelish.cs.man.ac.uk/rdf/species/Aptosyax/grypus>  
<http://www.w3.org/2004/02/skos/core#exactMatch>  
<http://dbpedia.org/resource/Giant_Salmon_Carp>.
```

```
<http://fishdelish.cs.man.ac.uk/rdf/species/Aptosyax/grypus>  
<http://www.w3.org/2004/02/skos/core#exactMatch>  
<http://lod.taxonconcept.org/ses/RocJQ#Species>.
```

```
<http://fishdelish.cs.man.ac.uk/rdf/species/Aptosyax/grypus>  
<http://www.w3.org/2004/02/skos/core#closeMatch>  
<http://purl.uniprot.org/taxonomy/143610>.
```


Finally, the RDF files that contain the links were deployed in a triple store to allow the execution of SPARQL queries to consume this data with the application that is described in the next chapter.

3.6. The Fish Species Finder Application

The Fish Species Finder Application was created to demonstrate the use of RDF links in the development of tools that combine information from different resources.

The main purpose of this tool is to present basic information of fish species from the different data sources to compare it in a single web page. As discussed previously, some species information, such as taxonomic classification, authority or common name, might differ from one resource to another as a result of both biological and technological issues.

This application was developed in Java, using the Jena ARQ package, a query engine that enables the execution of SPARQL queries. The application consists of a set of JSP files and Java classes, and its architecture roughly follows the MVC (Model View Controller) model, where the presentation, navigation and data access are separated in different software components.



Fish Species Finder

Find a fish species

Enter the common or binominal name (genus and species) of a fish species:

Search type : Common name ☐ Binominal name ☐

Common name :


Genus :

Species :

Linking Fish Species Data Project | School of Computer Science | The University of Manchester

Figure 18: Fish Species Finder

The user can search for fish species using the common or binominal name, or a substring of any of these names. The application will search, with the execution of a SPARQL query, in the FishDelish dataset for the species that match the introduced string. Then, a list of all the entities found is presented. The species *Betta splendens* was searched as an example.



Fish Species Finder

Find a fish species

Enter the common or binominal name (genus and species) of a fish species:

Search type : Common name ☐ Binominal name ☒

Common name :

Genus :


Species :

URI	Genus	Species	Common Name
http://fishdelish.cs.man.ac.uk/rdf/species/Betta/splendens	Betta	splendens	Siamese fighting fish

Linking Fish Species Data Project | School of Computer Science | The University of Manchester

Figure 19: Fish Species Finder search

Afterwards, the user has to select the required element. The next screen presents the information from all the datasets that contain data about the selected species. The RDF links are used at this point to identify the data sources that contain data of the selected species. A SPARQL query is executed in the local triple store to find the links to the other resources. Then, for each found link, a query is executed in the corresponding endpoint of the external resource to retrieve further data of the species.



Fish Species Finder

Information about:
<http://fishdelish.cs.man.ac.uk/rdf/species/Betta/splendens>

[Back](#) [New Search](#)

Number of links to other datasets: 4

URI	Predicate
http://purl.uniprot.org/taxonomy/158456	http://www.w3.org/2004/02/skos/core#exactMatch
http://lod.taxonconcept.org/ses/bGPhI#Species	http://www.w3.org/2004/02/skos/core#exactMatch
http://dbpedia.org/resource/Siamese_fighting_fish	http://www.w3.org/2004/02/skos/core#exactMatch
http://bio2rdf.org/taxonomy:158456	http://www.w3.org/2004/02/skos/core#exactMatch

Species Nomenclature

Data Source	Scientific Name	Common Name	Authorship
FishDelish	Betta splendens	Siamese fighting fish	Regan, 1910
Taxonconcept	Betta splendens	Siamese Fighting Fish	Regan 1910
DBpedia	Betta splendens	Siamese fighting fish	C._Tate_Regan
Bio2RDF	Betta splendens	Siamese fighting fish	

Taxonomic Classification

Data Source	Kingdom	Phylum	Class	Order	Family	Genus
Taxonconcept	Animalia	Chordata	Actinopterygii	Perciformes	Osphronemidae	Betta
DBpedia	Animal	Chordata	Actinopterygii	Perciform	Osphronemidae	Betta

View more data from [FishDelish](#) [Taxonconcept](#) [DBpedia](#) [Uniprot](#) [Bio2RDF](#)

[Back](#) [New Search](#)

FISHDELISH

TAXONCONCEPT

UNIPROT TAXONOMY

BIO2RDF

DBPEDIA

Linking Fish Species Data Project | School of Computer Science | The University of Manchester

Figure 20: Fish Species Finder, *Betta splendens*

To exemplify the use of this application in the identification of disagreements in species information, the following screen capture is provided, where a disagreement in the authority of the species

Syngnathus pelagicus can be noted.

Number of links to other datasets: 1

URI	Predicate
http://lod.taxonconcept.org/ses/iDlqx#Species	http://www.w3.org/2004/02/skos/core#relatedMatch

Species Nomenclature

Data Source	Scientific Name	Common Name	Authorship
FishDelish	<i>Syngnathus pelagicus</i>	Sargassum pipefish	Linnaeus, 1758
Taxonconcept	<i>Syngnathus pelagicus</i>	Sargassum Pipefish	Risso 1810

Figure 21: Fish Species Finder, authority comparison

Additionally, this applications displays the data from the other resources that contain the selected species. The following screen capture provides the data of the species *Arripis trutta* from Taxonconcept.

Taxonconcept

Predicate	Object
http://lod.taxonconcept.org/ontology/txn.owl#kingdom	Animalia
http://purl.org/dc/terms/hasPart	http://lod.taxonconcept.org/ses/C6tttd#OriginalDescription
http://xmlns.com/foaf/0.1/depiction	http://upload.wikimedia.org/wikipedia/commons/4/42/9lbsalmon.jpg
http://lod.taxonconcept.org/ontology/txn.owl#hasNCBI	270544
http://lod.taxonconcept.org/ontology/txn.owl#hasSpeciesConceptID	urn:uuid:ab19b2d6-56a6-4dc7-a6d4-bb6cf21ecc97
http://www.w3.org/2004/02/skos/core#inScheme	http://lod.taxonconcept.org/ontology/txn.owl#TaxonConcept_Scheme
http://www.w3.org/2000/01/rdf-schema#label	Arripis trutta se:C6tttd
http://lod.taxonconcept.org/ontology/txn.owl#inCoLClass	http://lod.taxonconcept.org/ontology/phylo/CoL/CoL_2010_base.owl#Class_Actinopterygii
http://lod.taxonconcept.org/ontology/txn.owl#family	Arripidae
http://www.w3.org/2004/02/skos/core#scopeNote	The intent is to provide a URI map to related information and eventually provide information to help determine a given specimen is a good match to this species concept.
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://lod.taxonconcept.org/ontology/txn.owl#SpeciesConcept
http://www.w3.org/2004/02/skos/core#closeMatch	http://dbpedia.org/resource/Eastern_Australian_salmon
http://lod.taxonconcept.org/ontology/txn.owl#inDBpediaClade	http://dbpedia.org/ontology/Fish
http://lod.taxonconcept.org/ontology/txn.owl#speciesConceptBasedOn	http://lod.taxonconcept.org/ontology/txn.owl#ObjectiveSpeciesModel
http://lod.taxonconcept.org/ontology/txn.owl#inCoLKingdom	http://lod.taxonconcept.org/ontology/phylo/CoL/CoL_2010_base.owl#Kingdom_Animalia
http://purl.org/dc/terms/identifier	http://lod.taxonconcept.org/ses/C6tttd#Species
http://purl.uniprot.org/core/scientificName	Arripis trutta
http://www.w3.org/2007/05/powder-s#describedby	http://lod.taxonconcept.org/ses/C6tttd.rdf
http://lod.taxonconcept.org/ontology/txn.owl#inCoLOrder	http://lod.taxonconcept.org/ontology/phylo/CoL/CoL_2010_base.owl#Order_Perciformes
http://lod.taxonconcept.org/ontology/txn.owl#inCoLPhylum	http://lod.taxonconcept.org/ontology/phylo/CoL/CoL_2010_base.owl#Phylum_Chordata
http://lod.taxonconcept.org/ontology/txn.owl#hasScientificName	Arripis trutta
http://purl.org/dc/terms/modified	2011-08-10T18:18:22-0500
http://lod.taxonconcept.org/ontology/txn.owl#hasWikipediaArticle	http://en.wikipedia.org/wiki/Eastern_Australian_salmon
http://purl.org/dc/terms/description	A Linked Open Data resource for the species concept Arripis trutta se:C6tttd
http://lod.taxonconcept.org/ontology/txn.owl#genus	Arripis
http://lod.taxonconcept.org/ontology/txn.owl#order	Perciformes
http://lod.taxonconcept.org/ontology/txn.owl#class	Actinopterygii
http://purl.org/dc/terms/isPartOf	http://lod.taxonconcept.org/ontology/void#TaxonConcept
http://lod.taxonconcept.org/ontology/txn.owl#hasTaxonNameID	http://gmi.globalnames.org/name_strings/40370470-72b5-5855-bef1-0bccaddee738
http://lod.taxonconcept.org/ontology/txn.owl#thumbnail	http://assets.taxonconcept.org/wt/Actinopterygii/Eastern_Australian_salmon_01.jpg
http://lod.taxonconcept.org/ontology/txn.owl#phylum	Chordata
http://lod.taxonconcept.org/ontology/txn.owl#specificEpithet	trutta

Figure 22: Fish Species Finder, *Arripis trutta* Taxonconcept data

In summary, this application demonstrates the utility of RDF links for combining data from multiple resources, allowing the user to visualise, compare and analyse information from different origins.

4. CONCLUSIONS

In this dissertation it was described the procedure followed to create RDF links to connect FishDelish with other data sources, and an application was created to exemplify the use of these links. Several lessons and conclusions can be drawn from this work, which are discussed below.

Undoubtedly, the greatest challenge that must be faced for the creation of RDF links is the identification of the entities that must be linked. In the biodiversity domain, the lack of stable and universal identifiers makes it difficult the creation of accurate links of URI aliases.

The development of tools to perform this identification automatically is required usually, as a manual generation is not feasible for large amounts of data. However, the automation of this task can represent the risk of creating some incorrect links due to false matches, and the risk of not identifying some correct matches, as the recognition of aliases must be performed frequently with the use of approximate string matching algorithms, which do not guarantee total accuracy. The use of these types of algorithms to detect aliases will always add a degree of uncertainty about the quality of the links. Therefore, there is a trade-off between constructing the links accurately and doing it quickly with software tools.

The extraction of data from external data sources is another of the difficulties that must be overcome, because in some cases the presence of mechanisms to find and retrieve data is reduced.

The creation of links also depends on other factors that are not under our control. For instance, the availability of datasets of the domain of interest, the amount and quality of information of the external resources, and the availability of entities that overlap the base dataset.

The methodology used in this project may be applied under other circumstances to link data from other domains. In particular, the sequence of tasks that were executed can be adopted to link datasets from any domain, because it is not attached to any characteristic of the biodiversity domain. Moreover, the key component used in this project, the application to create the links, may be used in any scenario where a piece of data from the elements of the datasets to be linked can be compared to indicate if a link must be generated.

However, this application and its design cannot be used under circumstances where an exact or

approximate string comparison is not useful to identify the elements to be linked. This design would produce a high rate of incorrect links in domains with a large presence of homonyms, and would fail to link datasets in different languages or containing a large number of synonyms. To link these kind of datasets, other methodologies suggested in [19] [82] [83] might be used, which basically involves comparing several pieces of data and calculating a similarity measure involving all the pairs of data, comparing the structure of the graph instead of individual nodes, and using external thesaurus or dictionaries to detect similar entities.

The following advantages of the applied methodology can be identified:

- It is required the execution of only one query to retrieve the data from the target datasets.
- Datasets in any formats and file types can be processed.
- The generation of links and manipulation of data can be performed offline, once the datasets have been retrieved.

On the other hand, the method used in this project present some disadvantages:

- It is not convenient in cases where the datasets are small, because a manual procedure can be used, avoiding the technical difficulties and producing more accurate links.
- A software component must be developed for every dataset that must be processed.

Other procedures to construct the links may have been applied, such as the execution of federated queries. This approach eliminates the need of creating an application to discover the links, but presents the disadvantages of producing very complex queries and reducing the ability to manipulate the strings; probably it is not possible to implement a similarity string algorithm with SPARQL.

The Fish Species application was created to exemplify the utility of RDF links. Nevertheless, the output of this application could have been produced without the existence of the links, based on the identification of overlapping entities in real time. However, this alternative involves the execution of four more queries in different endpoints to find the URI aliases at runtime, which would reduce notably the performance of the application, compared to the execution of only one query in the local endpoint. With the use of links, the application can know if there is a link to, for example, DBpedia; if there is not such link, no query to DBpedia is executed. Besides, the comparison of the authority could not be performed in real time with the same accuracy, as it requires the aid of a human in some cases to decide if the strings being compared refer to the same authority or not.

Although this specific application may have been created without the use of RDF links (with the disadvantages that it represents), they are irreplaceable components for other types of applications that crawl the Web of Data by following links, and also help to make more visible a dataset within the Web of Data.

The following point that must be analysed is if a similar output can be produced without the use of Linked Data technologies. Supposing that the different data providers offer other services to find and retrieve data, such as APIs or web services, it can be produced. But the main difference between using these services and using Linked Data, is that with Linked Data we have access to a repository of data compared to a big database, and the access of data using APIs and web services is restricted according the methods provided; as a result, a database cannot be explored and manipulated with the same freedom offered by Linked Data, in addition to other disadvantages of APIs and web services that were discussed previously in this paper. Additionally, as the data that can be extracted using APIs or web services is not linked, the identification of records referring to the same species, if it is possible, must be performed in real time, with the disadvantages that it represents.

Thus, the following advantages of Linked Data were identified:

- It offers real access to a global data space.
- It makes use of standardised technologies.
- It is not necessary to learn a new API or other service to retrieve data from each resource.
- The data from heterogeneous resources can be integrated.
- The difficulties to develop tools to combine, compare and analyse data from different data sources are decreased.

The following problems with Linked Data were found:

- The vast majority of data providers that offer content on the Web have not adopted the Linked Data principles to publish data. Consequently, the availability of data in RDF format is reduced.
- It can be difficult to learn all the technologies and concepts related with Linked Data.
- The Linked Data infrastructure currently is neither robust nor reliable.

- There is a lack of a central repository of information about all the Linked Data resources and vocabularies.
- The use of heterogeneous identifiers across the different data sources to refer to the same objects.

Other conceptual issues with Linked Data can be identified. One of its main principles establishes the use of URIs to identify things from the real world. Then, it would be expected to have a unique URI for each entity; for instance, to have only one URI to represent the species *Salmo salar* or one URI to represent the city of Manchester. This architecture would solve the problems of integrating data from different resources, but, at the same time, would bring other problems and questions. For example, if a URI is dereferenced, what data should be presented? or if data from multiple providers is displayed, who should host the services for that URI?

In conclusion, the creation of RDF links represents a number of problems that can be solved with different approaches, but the selection of the most adequate depends on the characteristics of the domain of interest, and the level of accuracy required. Additionally, the use of links eases the creation of tools to combine data from heterogeneous resources. Those tools that combine data might be created without the use of Linked Data technologies, but the use of this technology enables their creation with less effort, compared to other current similar technologies. However, to expand the use of this technology, some issues must be solved.

The future work corresponds to the development of more reliable and simpler methodologies to discover links that can be applied under different domains and scenarios.

REFERENCES

- [1] T. Berners-Lee, "Linked Data - Design Issues," 27-Jul-2006. [Online]. Available: <http://www.w3.org/DesignIssues/LinkedData.html>. [Accessed: 26-Mar-2011].
- [2] T. Heath and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space," *Synthesis Lectures on the Semantic Web: Theory and Technology*, vol. 1, no. 1, pp. 1-136, Feb. 2011.
- [3] T. Heath, "How Will We Interact with the Web of Data?," *IEEE Internet Computing*, vol. 12, no. 5, pp. 88-91, Sep. 2008.
- [4] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data - The Story So Far," *International Journal on Semantic Web and Information Systems*, vol. 5, no. 3, pp. 1-22, 2009.
- [5] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, vol. 284, no. 5, pp. 34-43, May. 2001.
- [6] C. Becker and C. Bizer, "DBpedia Mobile: A Location-Enabled Linked Data Browser," *1st Workshop about Linked Data on the Web (LDOW2008)*, 2008.
- [7] C. Bizer, "The Emerging Web of Linked Data," *Intelligent Systems, IEEE*, vol. 24, no. 5, pp. 87-92, 2009.
- [8] F. Manola, E. Miller, and B. McBride, "RDF Primer. W3C Recommendation 10 February 2004," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/rdf-primer/>. [Accessed: 29-Apr-2011].
- [9] T. Segaran, C. Evans, and J. Taylor, *Programming the Semantic Web*, 1st ed. USA: O'Reilly Media, 2009.
- [10] D. Beckett and B. McBride, "RDF/XML Syntax Specification (Revised)," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/REC-rdf-syntax/>. [Accessed: 23-May-2011].
- [11] D. Brickley, R. V. Guha, and B. McBride, "RDF Vocabulary Description Language 1.0: RDF Schema," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/rdf-schema/>. [Accessed: 01-May-2011].
- [12] H. Knublauch et al., "A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.2," *The University Of Manchester*, Mar. 2009.
- [13] D. McGuinness and F. Van Harmelen, "OWL Web Ontology Language Overview," 18-Aug-2003. [Online]. Available: <http://www.w3.org/TR/2003/CR-owl-features-20030818/>. [Accessed: 01-May-2011].
- [14] S. Bechhofer et al., "OWL Web Ontology Language Reference," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/owl-ref/>. [Accessed: 03-May-2011].
- [15] S. Bechhofer and A. Miles, "SKOS Simple Knowledge Organization System Reference," 18-Aug-2009. [Online]. Available: <http://www.w3.org/TR/skos-reference/>. [Accessed: 01-May-2011].
- [16] E. Prud'hommeaux and A. Seaborne, "SPARQL Query Language for RDF," 15-Jan-2008. [Online]. Available: <http://www.w3.org/TR/rdf-sparql-query/>. [Accessed: 23-Aug-2011].
- [17] M. Grobe, "Introduction to SPARQL." Indiana University, 17-Feb-2011.
- [18] "openRDF.org: About Sesame." [Online]. Available: <http://www.openrdf.org/about.jsp>. [Accessed: 23-Aug-2011].
- [19] Y. Liu, F. Scharffe, and C. Zhou, "Towards practical rdf datasets fusion," in *Workshop on Data Integration through Semantic Technology (DIST2008)*, ASWC, 2008.

- [20] O. Hassanzadeh and M. Consens, "Linked movie data base," in *Proceedings of the 2nd Workshop on Linked Data on the Web (LDOW2009)*, 2009.
- [21] B. Behkamal, M. Kahani, S. Paydar, M. Dadkhah, and E. Sekhavaty, "Publishing Persian linked data; challenges and lessons learned," in *Telecommunications (IST), 2010 5th International Symposium on*, 2010, pp. 732-737.
- [22] P. Jain, P. Hitzler, P. Z. Yeh, K. Verma, and A. P. Sheth, "Linked Data is Merely More Data," *AAAI Spring Symposium 'Linked Data Meets Artificial Intelligence'*, AAAI, p. 82--86, 2010.
- [23] T. Heath, D. Ayers, Y. Raimond, and C. Bizer, "Interlinking Open Data on the Web," 2007.
- [24] "Linking Open Data." [Online]. Available: <http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>. [Accessed: 01-May-2011].
- [25] R. Cyganiak and A. Jentzsch, "The Linking Open Data cloud diagram," 22-Sep-2010. [Online]. Available: <http://richard.cyganiak.de/2007/10/lod/>. [Accessed: 09-May-2011].
- [26] "What is Freebase? - Freebase." [Online]. Available: http://wiki.freebase.com/wiki/What_is_Freebase%3F. [Accessed: 02-May-2011].
- [27] "GeoSpecies Knowledge Base." [Online]. Available: <http://about.geospecies.org/>. [Accessed: 02-May-2011].
- [28] B. S. Betts, F. McNamara, P. Sinclair, T. Scott, and Y. Raimond, "Case Study: Use of Semantic Web Technologies on the BBC Web Sites," *W3C Semantic Web Use Cases and Case Studies*. [Online]. Available: <http://www.w3.org/2001/sw/sweo/public/UseCases/BBC/>. [Accessed: 02-May-2011].
- [29] "DBpedia: About." [Online]. Available: <http://dbpedia.org/About>. [Accessed: 02-May-2011].
- [30] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," *Lecture Notes in Computer Science*, vol. 4825, pp. 722-735, 2007.
- [31] R. Froese and D. Pauly, "FishBase," 2011. [Online]. Available: <http://www.fishbase.org/home.htm>. [Accessed: 14-May-2011].
- [32] "Fishdelish : JISC." [Online]. Available: <http://www.jisc.ac.uk/whatwedo/programmes/inf11/jiscexpo/fishdelish.aspx>. [Accessed: 12-Mar-2011].
- [33] "The fishDelish Project," 2010. [Online]. Available: <http://fishdelish.cs.man.ac.uk/>. [Accessed: 24-Mar-2011].
- [34] "FISHlink Project Website," 2010. [Online]. Available: <http://www.fishlinkonline.org/>. [Accessed: 24-Mar-2011].
- [35] "Integrated Taxonomic Information System." [Online]. Available: <http://www.its.gov/>. [Accessed: 25-Mar-2011].
- [36] "Species 2000 website." [Online]. Available: <http://www.sp2000.org/>. [Accessed: 28-Apr-2011].
- [37] "Catalogue of Life." [Online]. Available: <http://www.catalogueoflife.org/>. [Accessed: 29-Jul-2011].
- [38] "Encyclopedia of Life." [Online]. Available: <http://www.eol.org/>. [Accessed: 04-Aug-2011].
- [39] "Encyclopedia of Life Blog » Species Pages." [Online]. Available: <http://blog.eol.org/category/species-pages/>. [Accessed: 04-Aug-2011].
- [40] "Global Biodiversity Information Facility." [Online]. Available: <http://www.gbif.org/>. [Accessed: 28-Jul-2011].

- [41] R.L.Kotpal, *Modern Text Book of Zoology: Vertebrates*, 3rd ed. New Delhi, India: Rastogi Publications, 2005.
- [42] C. Hickman, L. Roberts, S. Keen, D. Eisenhour, A. Larson, and H. l'Anson, *Integrated Principles of Zoology*, Fifteenth edition. New York, NY, USA: McGraw-Hill, 2011.
- [43] "FishWise Professional." [Online]. Available: <http://www.fishwisepro.com/about.aspx>. [Accessed: 24-Jul-2011].
- [44] W. N. Eschmeyer and R. Fricke, "Catalog of Fishes electronic version," 14-Jul-2011. [Online]. Available: <http://research.calacademy.org/ichthyology/catalog>. [Accessed: 05-Aug-2011].
- [45] J. L. Sumich and J. F. Morrissey, *Introduction to the biology of marine life*, Eighth edition. Canada: Jones & Bartlett Publishers, 2004.
- [46] M. Armstrong, *Aquatic Life of the World*, vol. 1. New York, NY, USA: Marshall Cavendish, 2001.
- [47] M. Bailey and P. Burgess, *Tropical Fishlopaedia: A Complete Guide to Fish Care*, First edition. Ringpress Books, 2000.
- [48] T. Dewey, "What is in a Scientific Name?," *The Animal Diversity Web, University of Michigan Museum of Zoology*, 2006. [Online]. Available: http://animaldiversity.ummz.umich.edu/site/animal_names/scientific_name.html. [Accessed: 25-Apr-2011].
- [49] R. D. M. Page, "Biodiversity informatics: the challenge of linking data and the role of shared identifiers," *Briefings in Bioinformatics*, vol. 9, no. 5, pp. 345 -354, 2008.
- [50] P. DeVries, "TaxonConcept Blog," *TaxonConcept - Species Concepts for the Semantic Web*. [Online]. Available: <http://www.taxonconcept.org/how-to/>. [Accessed: 02-May-2011].
- [51] "CKAN - the Data Hub." [Online]. Available: <http://ckan.net>. [Accessed: 02-May-2011].
- [52] "SWEO Community Project: Linking Open Data on the Semantic Web," *W3C*. [Online]. Available: <http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/DataSets>. [Accessed: 05-Jul-2011].
- [53] A. Hogan, A. Harth, J. Umbrich, S. Kinsella, A. Polleres, and S. Decker, "Searching and Browsing Linked Data with SWSE: the Semantic Web Search Engine," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. In Press, Accepted Manuscript.
- [54] T. Scott, "Wildlife Finder: David Attenborough's favourite moments and more," *BBC Internet Blog*, 29-Sep-2009. [Online]. Available: http://www.bbc.co.uk/blogs/bbcinternet/2009/09/wildlife_finder_david_attenbor.html. [Accessed: 14-Jul-2011].
- [55] "UniProtKB." [Online]. Available: <http://www.uniprot.org/help/uniprotkb>. [Accessed: 17-Jul-2011].
- [56] The UniProt Consortium, "Ongoing and future developments at the Universal Protein Resource," *Nucleic Acids Research*, vol. 39, no. Database, p. D214-D219, Nov. 2010.
- [57] European Environment Agency, "EUNIS biodiversity database." [Online]. Available: <http://eunis.eea.europa.eu/>. [Accessed: 20-Jul-2011].
- [58] "Fishes of Texas Project Documentation." [Online]. Available: <https://sites.google.com/site/fishesoftexasdocumentation/>. [Accessed: 22-Jul-2011].
- [59] Spire Research Group, "Spire - Applying semantic web technologies in research ecoinformatics." [Online]. Available: <http://spire.umbc.edu/>. [Accessed: 23-Jul-2011].
- [60] "Alexa Top 500 Global Sites." [Online]. Available: <http://www.alexa.com/topsites>. [Accessed:

15-Jul-2011].

- [61] "Wikipedia:About - Wikipedia, the free encyclopedia." [Online]. Available: <http://en.wikipedia.org/wiki/Wikipedia:About>. [Accessed: 25-Jul-2011].
- [62] "About Freebase." [Online]. Available: http://wiki.freebase.com/wiki/Main_Page. [Accessed: 31-Jul-2011].
- [63] "Freebase - Type - Animal," 09-Dec-2008. [Online]. Available: <http://www.freebase.com/schema/biology/animal>. [Accessed: 05-Apr-2011].
- [64] "OpenCyc.org." [Online]. Available: <http://www.opencyc.org/>. [Accessed: 01-Aug-2011].
- [65] J. Zaino, "OpenCyc Hooks Into Linked Data Web," *semanticweb.com*, 29-Sep-2008. [Online]. Available: http://semanticweb.com/opencyc-hooks-into-linked-data-web_b175. [Accessed: 09-Jun-2011].
- [66] F. Belleau, M.-A. Nolin, N. Tourigny, P. Rigault, and J. Morissette, "Bio2RDF: towards a mashup to build bioinformatics knowledge systems," *Journal of Biomedical Informatics*, vol. 41, no. 5, pp. 706-716, Oct. 2008.
- [67] P. Ansell et al., "Bio2RDF Network Of Linked Data," *Building*, 2008.
- [68] "PubMed," *US National Library of Medicine, National Institutes of Health*. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed>. [Accessed: 02-Jul-2011].
- [69] "Global Names Architecture." [Online]. Available: <http://www.globalnames.org/>. [Accessed: 09-Aug-2011].
- [70] D. S. M. C. Press and J. Leonard, *Systems engineering fundamentals*. DIANE Publishing, 2001.
- [71] E. E. Ruppert and R. S. Fox, *Seashore animals of the Southeast: a guide to common shallow-water invertebrates of the southeastern Atlantic Coast*, 1st ed. University of South Carolina Press, 1988.
- [72] "SchemaWeb - RDF Schemas Directory." [Online]. Available: <http://www.schemaweb.info/>. [Accessed: 03-May-2011].
- [73] "(LOV) Linked Open Vocabularies." [Online]. Available: <http://labs.mondeca.com/dataset/lov/>. [Accessed: 03-May-2011].
- [74] "Schema Cache." [Online]. Available: <http://schemacache.com/>. [Accessed: 05-Aug-2011].
- [75] K. Jacobson, Y. Raimond, and T. Gangler, "The Similarity Ontology." [Online]. Available: <http://kakapo.dcs.qmul.ac.uk/ontology/musim/0.2/musim.html>. [Accessed: 14-Aug-2011].
- [76] K. Jacobson, "More on the Similarity Ontology," *kurtisrandom*, 28-Oct-2009. [Online]. Available: <http://kurtisrandom.blogspot.com/2009/10/more-on-similarity-ontology.html>. [Accessed: 14-Aug-2011].
- [77] T. Inkster, "Biological Taxonomy Vocabulary 0.2 (Core)," 06-Oct-2008. [Online]. Available: <http://ontologi.es/biol/ns>. [Accessed: 02-Aug-2011].
- [78] B. Ferris and T. Inkster, "The Association Ontology Specification," 13-Sep-2010. [Online]. Available: <http://smiy.sourceforge.net/ao/spec/associationontology.html>. [Accessed: 04-Aug-2011].
- [79] P. A. V. Hall and G. R. Dowling, "Approximate String Matching," *ACM Computing Surveys*, vol. 12, no. 4, pp. 381-402, Dec. 1980.
- [80] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol. 33, no. 1, pp. 31-88, Mar. 2001.

- [81] "Apache Commons." [Online]. Available: <http://commons.apache.org/>. [Accessed: 22-Aug-2011].
- [82] Y. Raimond, C. Sutton, and M. S, "Automatic Interlinking of Music Datasets on the Semantic Web."
- [83] P. Hsiung, A. Moore, D. Neill, and J. Schneider, "Alias Detection in Link Data Sets," *Proceedings of the International Conference on Intelligence Analysis*, 2004.

BIBLIOGRAPHY

- [1] P. Jokinen, J. Tarhio, and E. Ukkonen, "A Comparison of Approximate String Matching Algorithms," *Software: Practice and Experience*, vol. 26, no. 12, pp. 1439-1458, Dec. 1996.
- [2] O. Hartig and A. Langeegger, "A Database Perspective on Consuming Linked Data on the Web," *Datenbank-Spektrum*, vol. 10, no. 2, pp. 57-66, Aug. 2010.
- [3] "DBpedia Examples using Linked Data and Sparql « 3kbo." [Online]. Available: <http://blog.3kbo.com/2008/08/11/dbpedia-examples-using-linked-data-and-sparql/>. [Accessed: 28-Mar-2011].
- [4] "DBpedia Mobile." [Online]. Available: <http://wiki.dbpedia.org/DBpediaMobile>. [Accessed: 28-Apr-2011].
- [5] O. Hartig, C. Bizer, and J. C. Freytag, "Executing SPARQL queries over the web of linked data," *The Semantic Web-ISWC 2009*, pp. 293-309, 2009.
- [6] M. Hausenblas, "Exploiting Linked Data to Build Web Applications," *IEEE Internet Computing*, vol. 13, no. 4, pp. 68-73, 2009.
- [7] "FISH.Link." [Online]. Available: <http://www.jisc.ac.uk/whatwedo/programmes/mrd/clip/fishlink.aspx>. [Accessed: 26-Mar-2011].
- [8] J. Goodwin, C. Dolbear, and G. Hart, "Geographical Linked Data: The Administrative Geography of Great Britain on the Semantic Web," *Transactions in GIS*, vol. 12, pp. 19-30, Dec. 2008.
- [9] D. Chudnov, "In Which 'Linked Data' Means 'A Better Web'," *Computers in Libraries*, vol. 29, no. 8, pp. 31-33, 2009.
- [10] Y. Raimond, C. Sutton, and M. Sandler, "Interlinking Music-Related Data on the Web," *Multimedia, IEEE*, vol. 16, no. 2, pp. 52-63, 2009.
- [11] J. Zhao, A. Miles, G. Klyne, and D. Shotton, "Linked data and provenance in biological data webs," *Briefings in Bioinformatics*, vol. 10, no. 2, pp. 139-152, Mar. 2009.
- [12] T. Heath, "Linked Data? Web of Data? Semantic Web? WTF?," *Tom Heath's Displacement Activities*, 02-Mar-2009. [Online]. Available: <http://tomheath.com/blog/2009/03/linked-data-web-of-data-semantic-web-wtf/>. [Accessed: 26-Mar-2011].
- [13] M. Rüther, T. Bandholtz, and A. Logean, "Linked Environment Data for the Life Sciences," *1012.1620*, Dec. 2010.
- [14] M. Grobe, "RDF, Jena, SparQL and the 'Semantic Web'," in *Proceedings of the ACM SIGUCCS fall conference on User services conference - SIGUCCS '09*, St. Louis, Missouri, USA, 2009, p. 131.
- [15] V. Lopez, A. Nikolov, M. Sabou, V. Uren, E. Motta, and M. d' Aquin, "Scaling Up Question-Answering to Linked Data," in *Knowledge Engineering and Management by the Masses*, vol. 6317, Springer Berlin / Heidelberg, 2010, pp. 193-210.
- [16] D. Benslimane, S. Dustdar, and A. Sheth, "Services Mashups: The New Generation of Web Applications," *Internet Computing, IEEE*, vol. 12, no. 5, pp. 13-15, 2008.
- [17] G. Correndo, M. Salvadores, I. Millard, H. Glaser, and N. Shadbolt, "SPARQL query rewriting for implementing data integration over linked data," in *Proceedings of the 1st International Workshop on Data Semantics - DataSem '10*, Lausanne, Switzerland, 2010, p. 1.
- [18] S. Bechhofer, R. Stevens, and S. Jupp, "SKOS with OWL: Don't be Full-ish!," *OWLED*, vol. 432, 2008.
- [19] A. Halevy, P. Norvig, and F. Pereira, "The Unreasonable Effectiveness of Data," *IEEE Intelligent*

Systems, vol. 24, no. 2, pp. 8-12, Mar. 2009.

[20] H. Halpin, P. Hayes, J. McCusker, and D. McGuinness, "When owl:sameAs Isn't the Same: An Analysis of Identity in Linked Data," *Lecture Notes in Computer Science*, vol. 6496, pp. 305-320, 2010.

[21] C. C. Marshall and F. M. Shipman, "Which semantic web?," in *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, New York, NY, USA, 2003, pp. 57-66.

[22] S. Bechhofer et al., "Why Linked Data is Not Enough for Scientists," in *e-Science (e-Science), 2010 IEEE Sixth International Conference on*, 2010, pp. 300-307.

Appendix A – The Linking Open Data cloud

