Bulgarian Academy of Sciences

Institute of Information and Communication Technologies
Department of Linguistic Modelling and Knowledge Processing

Pensoft Publishers

Summary of Dissertation

submitted towards the degree

"Ph. D."

in doctoral program "Informatics" (01.01.2012)
professional field 4.6 "Informatics and Computer Science"

The Open Biodiversity Knowledge Management System in Scholarly Publishing

Author:
M. Sci. Viktor Senderov

Academic adviser:
Prof. Dr. Lyubomir Penev
Academic consultant:
Assoc. Prof. Dr. Kiril Simov

Sofia, 2019 г.

Introduction

Importance of the topic

The desire for an integrated information system serving the needs of the biodiversity community dates at least as far back as 1985 when the Taxonomy Database Working Group (TDWG)—later renamed to Biodiversity Informatics Standards but retaining the abbreviation TDWG—was established[1]. In 1999, the Global Biodiversity Information Facility (GBIF) was created after the Organization for Economic Cooperation and Development (OECD) had arrived at the conclusion that "an international mechanism is needed to make biodiversity data and information accessible worldwide" (*What is GBIF?*). The Bouchout declaration (*Bouchout Declaration* 2014) crowned the results of the European Union–funded project *pro-iBiosphere* that lasted from 2012 to 2014 and was dedicated to the task of creating an integrated biodiversity information system. The Bouchout declaration proposes to make scholarly biodiversity knowledge freely available as Linked Open Data (LOD). A parallel process in the U.S.A. started even earlier with the establishment of the Global Names Architecture, GNA (Patterson et al., 2010; Pyle, 2016b).

In 2014, the Horizon 2020 BIG4 consortium was formed between academia and industry dedicated to advancing biodiversity science. The project's mission statement reads "BIG4—Biosystematics, Informatics and Genetics of the big 4 insect groups: training tomorrow's researchers and entrepreneurs" (University of Copenhagen et al., 2014). An important member of the consortium is the academic publishing house and software company, Pensoft Publishers. Pensoft publishes several dozen well-known open access taxonomic journals[2] and, as a signatory of the Bouchout declaration, was a prime candidate to push the vision for an Open Biodiversity Knowledge Management System (OBKMS) forward. The presented Ph.D. project is based at Pensoft Publishers and at the Institute of Information and Communication Technology (IICT) of the Bulgarian Academy of Sciences with the goal to follow through pro-iBiosphere's vision.

Previous work

Due to the interdisciplinary nature of the thesis, this section will focus on two areas: (a) knowledge bases and Linked Open Data and (b) biodiversity publishing.

[1] A webpage with the history of TDWG dating back to 1985 can be viewed under `http://old.tdwg.org/past-meetings/`; however, a lot of the links are unfortunately broken and the page needs some maintenance.

[2] For example, ZooKeys, PhytoKeys, MycoKeys, and Biodiversity Data Journal (BDJ).

Knowledge bases and Linked Open Data

We shall start by first introducing *knowledge bases* and *knowledge-based systems*. We use the two terms interchangeably but tend to write the longer variant, knowledge-based system, when we want to emphasize aspects of the knowledge base that are not related to the underlying facts store (database).

It is useful to form one's concept of knowledge-based systems both by looking at explicit definitions and by looking at several examples of knowledge bases in practice. The term was already being widely discussed by the 1980's (Jarke et al., 1989) and early nineties (Harris et al., 1993) and was understood to mean the utilization of ideas from both database management systems (DBMS) and artificial intelligence (AI) to create a type of computer system called *knowledge base management system* (KBMS). Harris et al., 1993 writes that the characteristics of a knowledge base management system are that it contains "prestored rules and facts from which useful inferences and conclusions may be drawn by an inference engine." We should note that the phrase "prestored rules" comes from the time of first-generation AI systems that were rule-based. Recently, there has been progress in incorporating statistical techniques into databases (Mansinghka et al., 2015); however, in this project we are working with the classical rule-based definition. In other words, a knowledge base is, in our understanding, a suitable database tightly integrated with a logic layer.

Another relatively more recent development in knowledge-based systems has been the application of the Linked Data principles (Heath and Bizer, 2011). In fact, most existing knowledge bases emphasize the community aspects of making data more interconnected and reusable. Examples include Freebase (Bollacker et al., 2008), which was recently incorporated in WikiData (Vrandečić and Krötzsch, 2014; Pellissier Tanon et al., 2016), DBPedia (Auer et al., 2007), as well as Wolfram|Alpha (*Wolfram|Alpha, Making the wolrd's knowledge computable*) and the Google Knowledge Graph (Singhal, 2012). What these systems have in common is that an emphasis is placed not only on the logic layer allowing inference but on a unified information space: these systems act as nexus integrating information from multiple places and they follow to various degrees the principles of Linked Open Data (LOD).

Linked Open Data (Heath and Bizer, 2011) is a concept of the Semantic Web (Berners-Lee et al., 2001), which, when applied properly, ensures that data published on the Web is reusable, discoverable, and most importantly ensures that pieces of data published by different entities can work together. We will discuss the Linked Data principles and their application to OpenBiodiv in detail in Chapter 3.

Leveraging these developments modern knowledge bases place a bigger emphasis on interlinking data rather than on developing a complex inference machinery. There has been critique of the idea of bundling logic in the database layer as such bundling leads to increased complexity (Barrasa, 2017). The critique can be summarized with two points. First, bundling the logic near the data (especially when it is excessive for the task at hand) can lead to drastic performance decreases[3]. Second, the developing of new techniques (e.g. machine learning) can make the existing deep logic layer obsolete. Our view is that data is the commodity which is much more valuable, and the inference strategy (be it a rule-based logic layer, or a statistical machine learning technique) can be replaced as computational science moves forward. These ideas lead to an interesting conundrum in the choice of a database technology discussed in the subsequent sections.

[3] We will compare the performance of the stronger Web Ontology Language (OWL) logic layer with a weaker RDF Schema (RDFS) logic layer in Chapter 3. Resource Description Format (RDF) is a data model for storing statements about things discussed later.

Finally, a knowledge-based system ultimately needs to include user-interface components (UI's) and application programming interfaces (API's) or an application layer. These serve as the point-of-contact between human and machine, or machine and machine and are crucial to the success of any such system.

Biodiversity publishing

In the biomedical domain there are well-established efforts to extract information and discover knowledge from literature (e.g. Rebholz-Schuhmann et al., 2005; Momtchev et al., 2009; Williams et al., 2012). The biodiversity domain, and in particular biological systematics and taxonomy (from here on in this thesis referred to as *taxonomy*), is also moving in the direction of semantization of its research outputs (Agosti, 2006; Patterson et al., 2006; Kennedy et al., 2005; Penev et al., 2010a; Tzitzikas et al., 2013). The publishing domain has been modeled through the Semantic Publishing and Referencing Ontologies, SPAR Ontologies (Peroni, 2014). The SPAR Ontologies are a collection of ontologies incorporating, amongst others, FaBiO, the FRBR-aligned Bibliographic Ontology (Peroni and Shotton, 2012), and DoCO, the Document Component Ontology (Constantin et al., 2016). The SPAR Ontologies provide a set of classes and properties for the description of general-purpose journal articles, their components, and related publishing resources. Taxonomic articles and their components, on the other hand, have been modeled through the TaxPub XML Document Type Definition (DTD)—also referred to loosely as XML schema—and the Treatment Ontologies (Catapano, 2010). While TaxPub is the XML-schema of taxonomic publishing for several important taxonomic journals (e.g. ZooKeys, PhytoKeys, Biodiversity Data Journal), the Treatment Ontologies are still in development and have served as a conceptual template for OpenBiodiv-O (discussed in Chapter 2).

Taxonomic nomenclature is a discipline with a very long tradition. It transitioned to its modern form with the publication of the Linnaean System (Linnaeus, 1758). Already by the beginning of the last century, there were hundreds of taxonomic terms in usage (Witteveen, 2015). At present the naming of organismal groups is governed by by the International Code of Zoological Nomenclature, ICZN (International Commission on Zoological Nomenclature, 1999) and by the International Code of Nomenclature for algae, fungi, and plants, Melbourne Code (**mcneill_international_2012**). Due to their complexity (e.g. ICZN has 18 chapters and 3 appendices), it proved challenging to create a top-down ontology of biological nomenclature. Example attempts include the relatively complete NOMEN ontology (Dmitriev and Yoder, 2017) and the somewhat less complete Taxonomic Nomenclatural Status Terms, TNSS[4].

There are several projects that are aimed at modeling the broader biodiversity domain conceptually. Darwin Semantic Web, Darwin-SW (Baskauf and Webb, 2016) adapts the previously existing Darwin Core (DwC) terms (Wieczorek et al., 2012) as Resource Description Framework (RDF). These models deal primarily with organismal occurrence data.

Modeling and formalization of the strictly taxonomic domain has been discussed by Berendsohn (Berendsohn, 1995) and later, e.g., in (Franz and Peet, 2009; Sterner and Franz, 2017). Noteworthy efforts are the XML-based Taxonomic Concept Transfer Schema (Taxonomic Names and Concepts Interest Group, 2006) and a now defunct Taxon Concept ontology. Very recently, the TDWG community has attempted to resurrect the Taxon Concept ontology with the Taxonomic Names and Concepts Interest

[4] Even though it is unknown to the authors whether TNSS was published in peer-reviewed literature, remnants of it can still be found on GitHub, e.g. under `https://github.com/pensoft/OpenBiodiv/blob/master/ontology/contrib/taxonomic_nomenclatural_status_terms.owl`.

Group. The group discussions can be accessed under `https://github.com/tdwg/tnc`. Interestingly the very first GitHub issue discussed OpenBiodiv-O and the possibility of its adoption as a TDWG standard.

By the time the OpenBiodiv project started in June 2015, a number of articles had been previously published on the topics of linking data and sharing identifiers in the biodiversity knowledge space (Page, 2008), unifying phylogenetic knowledge (Parr et al., 2012), taxonomic names and their relation to the Semantic Web (Page, 2006; Patterson et al., 2010), and aggregating and tagging biodiversity research (Mindell et al., 2011). Some partial discussion of OBKMS was to be found in the science blog iPhylo (Page, 2014, 2015). The legal aspects of the OBKMS had been discussed by Egloff et al., 2014.

Furthermore, several tools and systems that deal with the integration of biodiversity and biodiversity data had been developed by different groups. Some of the most important ones are UBio, Global Names, BioGuid, BioNames, Pensoft Taxon Profile, and the Plazi Treatment Repository[5].

Key findings

The key findings from the papers cited in the previous paragraphs can be summarized as follows:

1. Biodiversity science deals with disparate types of data: taxonomic, biogeographic, phylogenetic, visual, descriptive, and others. These data are siloed in unlinked data repositories.

2. Biodiversity databases need a universal system of naming concepts due to the inefficiencies of Linnaean names for modern taxonomy. Taxonomic concept labels have been proposed as a human-readable solution and stable globally unique identifiers of taxonomic concepts had been proposed as a machine-readable solution.

3. There is a base of digitized semi-structured biodiversity information online with appropriate licenses waiting to be integrated as a knowledge base.

Goal and objectives

Given the huge international interest in OBKMS, this dissertation started the OpenBiodiv project, the goal of which is to contribute to OBKMS by creating an open knowledge-based system of biodiversity information extracted from scholarly literature. In order to complete the system, the following objectives need to be achieved:

Objective 1: Architecture. Formally define OpenBiodiv as a knowledge-based system and create its integrated software architecture.

Objective 2: Ontology. Study the domain of biodiversity informatics and biodiversity publishing and develop an ontology allowing data integration from diverse sources.

[5] UBio: http://ubio.org/; Global Names: http://globalnames.org/; BioGuid: http://bioguid.org/; BioNames: http://bionames.org/; Pensoft Taxon Profile: http://ptp.pensoft.eu/; Plazi Treatment Repository: http://plazi.org/wiki/.

Objective 3: Linked open dataset. Create a Linked Open Dataset (LOD) on the basis of published taxonomic articles using the ontology defined in Objective 2.

Objective 4: Library. Develop methods for converting taxonomic publications into the semantic model of the ontology in order to support Objective 3.

Objective 5: Workflows. Develop practical workflows for continuously converting taxonomic data into taxonomic publications and thus updating the LOD dataset.

Objective 6: Web portal. Create a web-portal and example applications on top of the knowledge base.

Methodology

This dissertation has a methods and tools orientation: i.e. its goal is not the testing of particular scientific hypothesis but rather the theoretical design and practical implementation of a knowledge-management system. In this section I shall outline the "meta-choices" that I have made—such as what programming and database paradigms to use—before the design and implementation phase.

Choice of database paradigm for OpenBiodiv

We specify OpenBiodiv as a knowledge-based system with a focus on structuring and interlinking biodiversity data. Two of the possible database technologies that fit this requirement are semantic graph databases (triple stores) such as GraphDB (Ontotext, 2018) and labeled property graphs such as Neo4J (Neo4J Developers, 2012). Semantic graph databases offer a very simple data model: every fact stored in such a database is composed as a triple of *subject*, *predicate*, and *object*. Subjects of triples are always resource identifiers, whereas objects can be other resource identifiers or literal values (e.g. strings, numbers, etc.). Links between resources or between resources and literals are given by the predicates (also specified as identifiers). These links are sometimes referred to as *properties*. Thus, one can visualize a graph whose vertices are the objects or subjects given by resource identifiers or literals and whose edges are predicates.

Semantic graph databases have the unique feature that the logic layer is also expressed as triples stored in the database. This logic layer, known as *ontology*, is not only responsible for drawing conclusions from the data (inference), but also specifies the semantics of how knowledge should be expressed.

Labeled property graphs, on the other hand, offer a freer data model by allowing the edges of the knowledge graph to have properties as well. For example, in a labeled property graph whose vertices are two cities A and B and are connected by a property-predicate *connected by road*, it is possible to additionally attach the value “500 km” to that property. Thus, we indicate that the length of the road connecting the cities is 500 km.

Note that labeled property graphs are not any more expressive than what can be achieved by triples alone. In fact, complex relationships in a simple triple store can be expressed by making relationships into nodes that have properties on their own. This process is known as *reification*. For example, the two cities A and B can connect to a further vertex, R indicating the road. R will then have three properties: *start*, *end*, and *length*. The value (object) of *start* will be A, of *end* will be B, and of length will be the literal “500 km.”

Table 1: Differences between semantic graph databases (e.g. GraphDB) and labeled property graphs (e.g. Neo4j).

Criterion	Semantic database	Labeled property graph
Semantics	Stored in the database itself as OWL or RDFS statements. Provides a uniform data space. Requires expert ontologists to extract knowledge.	Formal semantics usually are missing. Quick deployment. Uniform data space harder to achieve.
Inference	Provided by the database itself from its ontology or expressed as SPARQL queries. General purpose, slower.	External to the database. Needs to be written for every specific task. Special purpose. Faster.
Community	Has a rich and mature community of ontologists and knowledge engineers. Lots of domain ontologies. Designed for inter-operability. Standards-driven.	Data models are created ad-hoc by data scientists or programmers for a particular task. Inter-operability requires effort and not of primary concern. Applications-driven.

We have summarized the differences between labeled property graphs and semantic graph databases in Table 1. After careful considerations, we settled on the triple store, i.e. semantic graph database as a choice of database technology. This decision was informed by the wide availability of high-quality ontologies and Resource Description Framework (RDF) data models in our domain (Baskauf and Webb, 2016; Peroni, 2014) and the popularity of the Semantic Web (Berners-Lee et al., 2001) in the community. Furthermore, our base at a publisher was more suited to a standards-driven foundational project as opposed to a particular application.

However, we believe that labeled property graphs are a freer and a more natural data model and are perfectly suited for biodiversity informatics. In particular they provide a much more natural formalism for relationships between taxonomic concepts (discussed in Chapter 2). Also, non-RDF semantic databases such as WikiData are gaining in popularity. Therefore, we believe that the applicability of RDF triple stores for OpenBiodiv should constantly be reëvaluated.

Choice of information sources

According to *pro-iBiosphere project final report* 2014, biodiversity and biodiversity-related data have two different "life-cycles." In the past, after an observation of a living organism had been made, it was recorded on paper and then the observation record was published in paper-based form. In order for biodiversity data to be available to the modern scientist, efforts are made nowadays to digitize those paper-based publications by Plazi Agosti et al., 2007 and the Biodiversity Heritage Library (Miller et al., 2012). For this purpose, several dedicated XML schemas have been developed (see Penev et al., 2011 for a review), of which TaxPub (Catapano, 2010) and TaxonX seem to be the most widely used (Penev et al., 2012). The digitization of publications contains several steps. After scanning and optical character recognition (OCR), text mining is combined with searching for particular kinds of data. This procedure leaves a trace in the form of marked-up (tagged) elements that can then be extracted and made available for future use and reuse (Miller et al., 2015).

In present day, biodiversity data and publications are mostly "born digital" as semantically Enhanced Publications (EP's, Claerbout and Karrenbach, 1992; Godtsenhoven et al., 2009; Shotton, 2009). According to Claerbout and Karrenbach, 1992, "an EP is a publication that is enhanced with research data, extra materials, post publication data and database records. It has an object-based structure with explicit links between the objects. An object can be (part of) an article, a data set, an image, a movie, a comment, a module or a link to information in a database." Semantically enhanced publications are thus natives of the Web and the Semantic Web unlike their paper-based predecessors.

The act of publishing in a digital, enhanced format, differs from the ground up from a paper-based publication. The main difference is that a digitally-published document can be structured in such a format as to be suitable both for machine processing and to the human eye. In the sphere of biodiversity science, Pensoft journals such as ZooKeys, PhytoKeys, and the Biodiversity Data Journal (BDJ) already function by providing EP's (Penev et al., 2010b).

Given the fact that Pensoft Publishers' and Plazi's publications cover a large part of taxonomic literature both in volume and also in temporal span, and the fact that the publications of those two publishers are available as semantic EP's, we've chosen Pensoft's journals and Plazi's treatments as our main sources of information.

Furthermore, we incorporate the taxonomic backbone of GBIF GBIF Secretariat, 2017a as a source for data integration. This is further discussed in Chapter 3.

Choice of development methodology and programming environment

In 2016, based on the outcomes of pro-iBiosphere and on the previous work in the area of biodiversity informatics, we published the Ph.D. plan for this research (Senderov and Penev, 2016). This publication can be considered as the first design specification of OpenBiodiv. However, in the course of developing the system, its design was changed iteratively through a feedback loop from collaborators from the BIG4 project[6] and various international collaborators. We view this positively and in the spirit of both *open science* and *agile software development* (Beck et al., 2001). This iterative approach differs from the waterfall approach where after a through design phase, the specifications "are frozen" and a lengthy implementation phase.

In recent years, the R programming language has been used widely in the field of data science (R Core Team, 2016). R has a rich library of software packages including such for processing XML (Wickham et al., 2018), for accessing rest API's (Wickham, 2017), and focuses on open science (Boettiger et al., 2015). The capabilities of R as function-oriented and interpreted language allow the iterative software development approach outlined in the previous paragraph to proceed rapidly. Furthermore, R is widely adopted in the biodiversity informatics community. For this reason, the R software environment was chosen as the main programming environment.

[6] The Ph.D. candidate, Viktor Senderov, is part of the Marie Skłodowska-Curie BIG4 International Training Network: Biosystematics, informatics and genomics of the big 4 insect groups: training tomorrow's researchers and entrepreneurs.

Open Science and The Semantic Web

After having specified the desired design and given the programming language, R, I would like to discuss some methodologies and frameworks that have been adopted to be more efficient, open, and reproducible.

I believe that OpenBiodiv needs to be addressed from the point of view of *Open Science*. According to Kraker et al., 2011 and to *Was ist Open Science?*, the six principles of open science are: open methodology, open source, open data, open access, open peer review, and open educational resources. It is my belief that the aim of open science is to ensure access to the whole research product: data, discoveries, hypotheses, and so on. This opening-up will ensure that the scientific product is reproducible and verifiable by other scientists (Mietchen, 2014). There is a very high interest in development of processes and instruments enabling reproducibility and verifiability, as can be evidenced for example by a special issue in Nature dedicated to reproducible research (*Challenges in irreproducible research* 2010). Therefore, the source code, data, and publications of OpenBiodiv will be published openly.

Moreover, OpenBiodiv should be thought of as integral part of the Semantic Web (Berners-Lee et al., 2001). The Semantic Web is a vision for the future of the web where not only documents but also data are connected.

Structure of the thesis

So far the raison d'être of the system and this thesis and an outline of its goal and objectives have been given in this Introduction. In Chapter 1, a formal specification and design of the desired system as well as an outline of its architecture will be presented; this chapter forms Objective 1. The subsequent chapters discuss the implementation of OpenBiodiv. Chapter 2 gives a conceptualization of the domain of scientific taxonomic publishing formalizes it by introducing the central result of this thesis, the ontology of OpenBiodiv (OpenBiodiv-O) and thus forms Objective 2. Chapter 3 describes the Linked Open Dataset that has been generated based on OpenBiodiv-O and forms Objective 3. Chapter 4 describes in detail the RDF4R software package (an R package for working with RDF), which was used to create the Linked Open Data (OpenBiodiv-LOD) and forms Objective 4. In Chapter 5, two case-studies for importing data into OpenBiodiv from important international repositories are discussed and thus it forms Objective 5. Chapter 6 discusses the website that has being prepared to serve on top of OpenBiodiv-LOD and its applications (Objective 6). In the Conclusion, I will explain how the results have been published and summarize the main results.

Chapter 1

Summary of Chapter 1: Architecture of OpenBiodiv

In this chapter, we provide the architectural blueprint, i.e. the specification and design of OpenBiodiv. We break up OpenBiodiv into components that will be treated in detail in subsequent chapters. We describe how these components inter-operate in order to form the OpenBiodiv knowledge-based system.

1.1 What is OpenBiodiv?

The understanding of OpenBiodiv as a knowledge-based system can be summarized as follows: OpenBiodiv is a database of interconnected biodiversity information together with logic and application layers allowing users to not only query the data but also discover additional facts of relevance implied by the data. The primary sources of information in OpenBiodiv are the journals of the academic publisher Pensoft, taxonomic information from Plazi, and the taxonomic backbone of Global Biodiversity Information Facility (GBIF).

The research problem of OpenBiodiv's architecture can be postulated as designing an open-access semantic RDF graph database, incorporating information stored in Pensoft, Plazi, and GBIF, and allowing the users of the system to ask complicated queries.

OpenBiodiv consists of (1) a semantic graph database, (2) a back-end code base, and (3) a front-end in the form of a web-portal facilitating the access to the underlying knowledge base (Fig. 1.1). OpenBiodiv enables the flow of information between international repositories for biodiversity data to Biodiversity Data Journal (BDJ) and other journals that use the ARPHA-BioDiv toolkit (Penev et al., 2017). As a second step, knowledge is extracted from such journals taking advantage of the TaxPub Document Type Definition (DTD)[1] introduced by Catapano, 2010. Example journals include ZooKeys, Biodiversity Data Journal (BDJ), PhytoKeys, MycoKeys, and so on[2]. At the same time, knowledge is extracted from Plazi TreatmentBank, an archive of legacy biodiversity literature containing over 200 thousand treatments[3] and updated every day. Last but not least, these sources are interlinked via GBIF's taxonomic backbone (GBIF Secretariat, 2017a). The extracted knowledge is then stored in a semantic graph database (Fig. 1.2).

[1] We will take the liberty and refer to TaxPub as an XML schema in the rest of the chapter.

[2] The journals can be accessed under `https://pensoft.net/browse_journals`.

[3] A treatment is a special section in a biological publication describing and discussion a species or a higher taxon. TreatmentBank is accessible under `https://http://plazi.org/resources/treatmentbank/`.

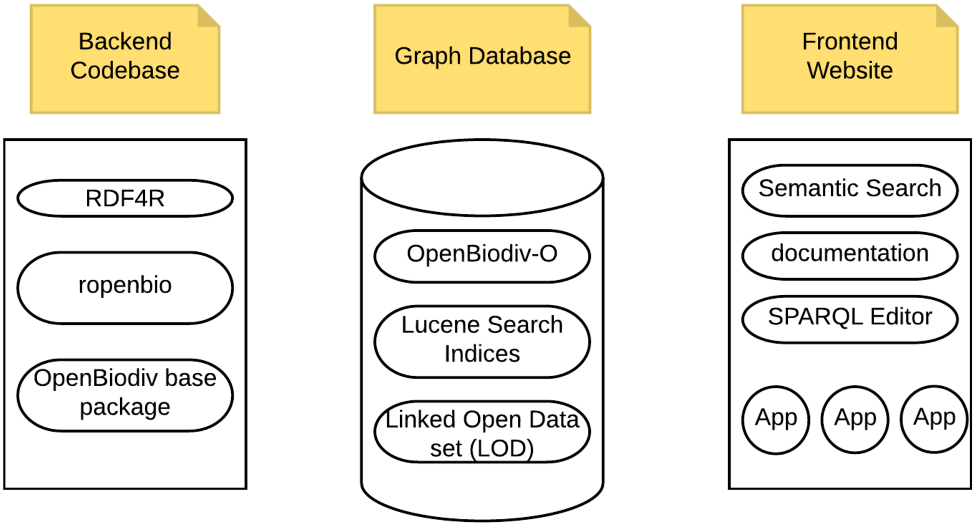


FIGURE 1.1: The components of OpenBiodiv.

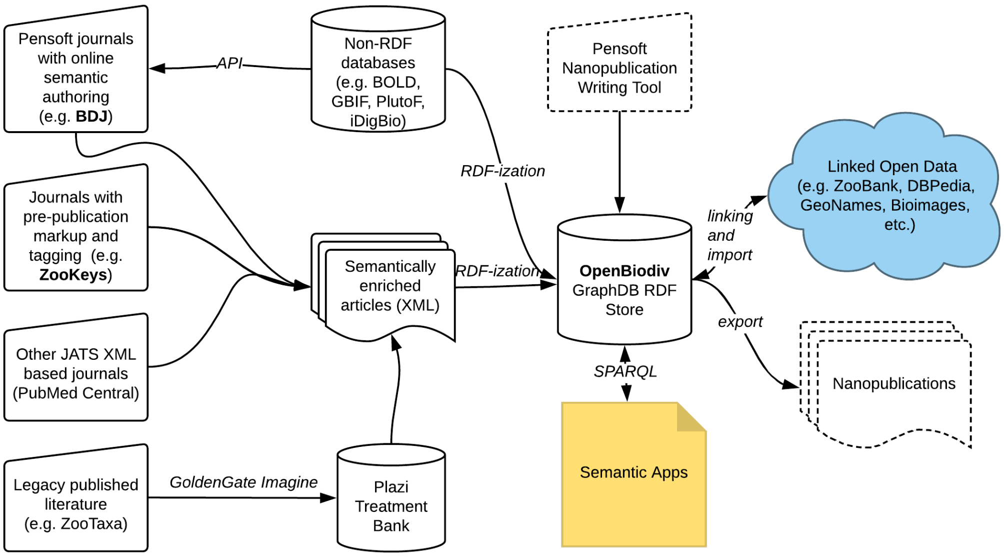

Pensoft journals with online semantic authoring (e.g. BDJ)
API
Non-RDF databases (e.g. BOLD, GBIF, PlutoF, iDigBio)
Pensoft Nanopublication Writing Tool
Journals with pre-publication markup and tagging (e.g. ZooKeys)
RDF-ization
Linked Open Data (e.g. ZooBank, DBPedia, GeoNames, Bioimages, etc.)
linking and import
Semantically enriched articles (XML)
RDF-ization
OpenBiodiv GraphDB RDF Store
export
Other JATS XML based journals (PubMed Central)
SPARQL
Nanopublications
Legacy published literature (e.g. ZooTaxa)
GoldenGate Imagine
Plazi Treatment Bank
Semantic Apps

FIGURE 1.2: Flow of information in the biodiversity data space until it reaches the OpenBiodiv semantic database. Dashed lines are components that have not been implemented yet.

1.2 Semantic graph database

A primary output of the OpenBiodiv effort is the creation of a semantic database based on knowledge extracted from the archives of Pensoft and Plazi and GBIF's taxonomic backbone and accessible under `http://graph.openbiodiv.net/`. A discussion of the components of the database follows.

1.2.1 OpenBiodiv ontology (OpenBiodiv-O)

The central result of the OpenBiodiv effort is the creation of a formal domain model of biodiversity publishing, the ontology OpenBiodiv-O (Senderov et al., 2017). The source code of the ontology and accompanying documentation can be accessed under `https://github.com/vsenderov/openbiodiv-o`. A detailed discussion is presented in Chapter 2.

1.2.2 OpenBiodiv Linked Open Dataset (OpenBiodiv-LOD)

Using OpenBiodiv-O and the infrastructure described later in this chapter a dataset incorporating approximately 200 thousand Plazi treatments, five thousand Pensoft articles, as well as GBIF's taxonomic backbone (over a million names) has been created. The dataset is available online through the workbench of the semantic database `http://graph.openbiodiv.net`. It is discussed in detail in Chapter 3.

1.3 Backend

In order to populate a semantic database it is necessary to create the infrastructure that converts raw data (text, images, data tables, etc.) into a structured semantic format allowing the interlinking of resource identifiers and the answering of complex queries. OpenBiodiv creates new infrastructure and extends existing infrastructure for transforming biodiversity scholarly publications into Resource Description Format (RDF) statements with the help of the components described in this section.

1.3.1 RDF4R: R package for working with RDF

One of the greater technical challenges for OpenBiodiv is the transformation of biodiversity information (e.g. taxonomic names, paper metadata, figures, etc.) stored as semi-structured XML into fully-structured semantic knowledge in the form of RDF. In order to solve this challenge, an R package has been developed that enables the creation, manipulation, and submission and retrieval to and from a semantic database of RDF statements. This package is accessible under an open source license on GitHub under `https://github.com/vsenderov/rdf4r`. We describe the package in Chapter 4.

1.3.2 OpenBiodiv Base and ROpenBio

In combination with the RDF4R package, the code-base is completed by one more R package, `ropenbio` and a code-base (OpenBiodiv Base) of scripts and documentation necessary to bootstrap the database. `ropenbio` utilizes the RDF4R package to convert semi-structured XML to RDF. It contains the "mappings" necessary for that conversion. It is available under `https://github.com/pensoft/ropenbio`. OpenBiodiv Base coordinates the invocation of `ropenbio`, contains scripts for the automatic import of new resources, and other housekeeping details. It is available under https://github.com/pensoft/openbiodiv. Their usage to generate the OpenBiodiv-LOD is discussed in Chapter 3.

1.3.3 Workflow for converting ecological metadata to a manuscript

Ecological Metadata Language (EML) is a popular format for describing ecological datasets (Michener et al., 1997). Biodiversity repositories such as GBIF and DataOne make use of this format to describe the datasets that they store. An import pipeline for importing an EML file as a BDJ data paper[4] has been developed as part of OpenBiodiv (Senderov et al., 2016). We describe this workflow in detail in Chapter 5. To access the pipeline interactively, go to https://arpha.pensoft.net, login to the system (registration is free), select “Start a new manuscript,” scroll all the way down to “Import a manuscript,” and follow the necessary steps to upload an EML and use it as a template for your new manuscript.

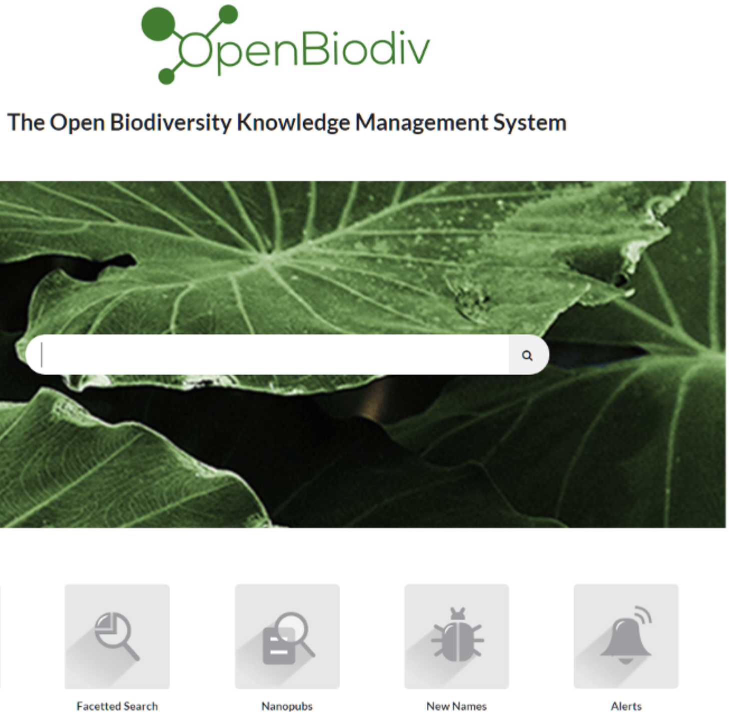
1.3.4 Workflow for importing specimen data into Biodiversity Data Journal

One of the important types of biodiversity data is occurrence data—data that documents the presence of a properly taxonomically identified organism at a given location and time. Such data is stored at international repositories such as BOLD, GBIF, PlutoF, and iDigBio. In order to facilitate data publishing, as well as to act as an entry point into OpenBiodiv, a pipeline for importing any occurrence record from these databases into a BDJ taxonomic paper has been developed (Senderov et al., 2016). We describe this workflow in detail in Chapter 5. To access the workflow interactively, go to https://arpha.pensoft.net, login to the system (registration is free), select "Start a new manuscript," select "Biodiversity Data Journal" as a journal and "Taxonomic Paper" as paper-type and "Create a manuscript." Then, in your new manuscript, expand the "Taxon treatments" section by clicking on the + sign next to it, give a test classification to your treatment (e.g. Animalia), click “Save” and you will be presented with a choice of subsections. Click the “Materials” section on the left to visualize the workflow. Look at the lower-part of the dialog, where “You may place multiple ID's...”—this is the part where you select external resource identifiers to be imported to your article.

1.4 Frontend

In addition to providing a searchable database endpoint, a website allowing semantic search and containing specific tasks packaged as apps is being developed (http://openbiodiv.net). The development of the site extends beyond the scope of the dissertation thesis and is driven by the Pensoft development team. A beta version is already operational Fig. 1.3. A limited discussion is found in Chapter **??**.

[4] A data paper (Chavan and Penev, 2011) is a paper in a scholarly (peer-reviewed) journal discussing a scientific dataset.



OpenBiodiv

The Open Biodiversity Knowledge Management System

FIGURE 1.3: Beta version of the OpenBiodiv website together with sample app icons.

1.5 IT

The system is deployed on a Debian GNU+Linux virtual machine. GraphDB runs with a 20 GB heap file and with the RDFS-Plus Optimized rule set[5]. Continuous operation is ensured by the automatic execution of scripts from the `run` directory of OpenBiodiv Base.

[5] This is necessitated by the fact that we reached a performance bottleneck the OWL inference. Discussed in Chapter 3

Chapter 2

Summary of Chapter 2: The OpenBiodiv Ontology

OpenBiodiv lifts biodiversity information from scholarly publications and academic databases into a computable semantic form. In this chapter, we introduce OpenBiodiv-O (Senderov et al., 2018), the ontology forming the knowledge and inferencing model of OpenBiodiv. OpenBiodiv-O provides a conceptual model of the structure of a biodiversity publication and the development of related taxonomic concepts. We first introduce the modeled domain in Domain Conceptualization and then formalize it in Results.

By developing an ontology focusing on biological taxonomy, our intent is to provide an ontology that fills in the gaps between ontologies for biodiversity resources such as Darwin-SW and semantic publishing ontologies such as the ontologies comprising the SPAR Ontologies. We take the view that it is advantageous to model the taxonomic process itself rather than any particular state of knowledge.

The source code and documentation are available under the CC BY license[1] from GitHub[2]. We start by introducing the domain of biological taxonomy and the related biodiversity sciences.

2.1 Domain Conceptualization

We give an introduction of the history of modern biological taxonomy starting with Carl Linnaeus (1707-1778) who proposed the modern organism grouping of *kingdoms, classes, orders, genera* and the usage of Latin binomial names in *Systema Naturae* (Linnaeus, 1758). We emphasize that the work of taxonomists to describe and organize biodiversity is far from complete. This informs the creation of our ontology not as a static formalization of the existing biological taxonomy in computer-readable form, but as a formalization of the *scientific process of biological taxonomy.*

We then describe in the detail what the scientific process of biological taxonomy entails. We start by introducing taxonomic concepts and how they are formed. A taxonomic concept is a scientific hypothesis (Deans et al., 2012) that a certain well-defined group of organisms exists in Nature. It is formed by examining specimens and necessarily entails a scientific grouping criterion, often called a species concept (Mallet, 2001; not to be confused with taxonomic concept!). Historically, organisms have been grouped by their appearance (morphological species concept) or reproductive behavior (biological species concept), but recently the focus has shifted towards grouping based on genetic relatedness (phylogenetic and genomic species concepts).

[1] Creative Commons Attribution 4.0 International Public License.

[2] https://github.com/vsenderov/openbiodiv-o/blob/master/LICENSE.md

We then describe the ranks of biological taxonomy and how they are regulated by International Codes **mcneill_international_2012**; International Commission on Zoological Nomenclature, 1999). The codes govern lower ranks: species, genus, family, order; higher ranks (e.g. phylum, kingdom, domain, etc.) are however free to be used by researchers as they view fit. This leads to multiple competing viewpoints.

Publishing taxonomic concepts is an integral step in the scientific workflow of every taxonomist. We describe the structure and types of taxonomic publications with a particular emphasis on the Treatment section. A Treatment is the section in a taxonomic publication where a taxonomic concept is circumscribed.

Previous work

We discuss previous efforts made to ontologize scientific publications and biological information. Particularly important are the Semantic Publishing and Referencing Ontologies (SPAR Ontologies, Peroni, 2014) and the TaxPub XML Document Type Definition ((Catapano, 2010) referred to loosely as XML schema). The modeling of biodiversity information is primarily influenced by the Codes (**mcneill_international_2012**; International Commission on Zoological Nomenclature, 1999), that were mentioned in the previous section, and by a variety of standards (e.g. Darwin Core, DwC, Wieczorek et al., 2012), published by the TDWG community.

Finally, we discuss the emerging field of concept taxonomy (Berendsohn, 1995; Franz and Peet, 2009; Sterner and Franz, 2017)—a re-imagination of how the circumscription process in biological taxonomy ought to work.

2.2 Methods

OpenBiodiv-O is expressed in Resource Description Framework (RDF). At the onset of the project, a consideration was made to use RDF in favor of a more complex data model such as Neo4J's (Senderov and Penev, 2016). The choice of RDF was made in order to be able to incorporate the multitude of existing domain ontologies into the overall model.

To develop the conceptualization of the taxonomic process and then the ontology we utilized the following process: (1) domain analysis and identification of important resources and their relationships; (2) analysis of existing data models and ontologies and identification of missing classes and properties for the successful formalization of the domain.

The formal structure of the ontology is specified by employing the RDF Schema (RDFS) and the Web Ontology Language (OWL). It is encoded as a part of a literate programming (Knuth, 1984) document in RMarkdown format titled "OpenBiodiv Ontology and Guide"[3]. The statements have been extracted from the RMarkdown file via *knitr* and are provided here as an appendix. It is also possible to request the ontology via Curl from its endpoint with the indication of `content-type: application/rdf+xml`. The vocabularies can be found as additional appendices, Taxonomic Statuses and RCC-5, and on the GitHub page[4].

A dataset (OpenBiodiv-LOD, will be described in detail in the next Chapter) from Pensoft's journals, Plazi's treatments, and GBIF's taxonomic backbone has been generated with OpenBiodiv-O and can be found at the SPARQL Endpoint [5]. The

[3] http://openbiodiv.net/ontology

[4] https://github.com/vsenderov/openbiodiv-o

[5] http://graph.openbiodiv.net/

endpoint is also accessible from the website[6], under "SPARQL Endpoint." Demos are available as "Saved Queries" from the workbench.

2.3 Results

We understand OpenBiodiv-O to be the *shared formal specification of the conceptualization* (Gruber, 1993; Obitko, 2007; Staab and Studer, 2009) that we have introduced in Background. OpenBiodiv-O describes the structure of this conceptualization, not any particular state of it.

There are several domains in which the modeled resources fall. The first one is the scholarly biodiversity publishing domain. The second domain is that of taxonomic nomenclature. The third domain is that of broader taxonomic (biodiversity) resources (e.g. taxonomic concepts and their relationships, species occurrences, traits). To combine such disparate resources together we rely on SKOS Miles and Bechofer. Unless otherwise noted, the default namespace of the classes and properties for this paper is `<http://openbiodiv.net/>`. The prefixes discussed here are listed at the beginning of the ontology source code.

2.3.1 Semantic Modeling of the Biodiversity Publishing Domain

We extend the framework of the SPAR Ontologies by introducing a new class for taxonomic articles, its subsections, as well as a new class for the mentioning of a taxonomic name (see next subsection) in an article. These new classes are summarized in Table 2.1.

TABLE 2.1: New biodiversity publishing classes introduced.

Class QName	Comment
`:Treatment`	section of a taxonomic article
`:NomenclatureSection`	subsection of Treatment
`:NomenclatureHeading`	contains a nomenclatural act
`:NomenclatureCitationList`	list of citations of related concepts
`:MaterialsExamined`	list of examined specimens
`:BiologySection`	subsection of Treatment
`:DescriptionSection`	subsection of Treatment
`:TaxonomicKey`	section with an identification key
`:TaxonomicChecklist`	section with a list of taxa for a region
`:TaxonomicNameUsage`	mention of a taxonomic name

The classes from this subsection are based on the TaxPub XML Document Type Definition (DTD, also referred to loosely as XML schema, Catapano, 2010), on the structure of Biodiversity Data Journal's taxonomic paper (Smith et al., 2013), and and on the Treatment Ontologies (Catapano and Morris, 2016).

Furthermore, we introduce two properties: *contains* (`:contains`) and *mentions* (`:mentions`). *contains* is used to link parts of the article together and *mentions* links parts of the article to other concepts.

A graphical representation of the relationships between instances of the publishing-related classes that OpenBiodiv introduces is to be found in the diagram in Fig. 2.1.

[6] http://openbiodiv.net/

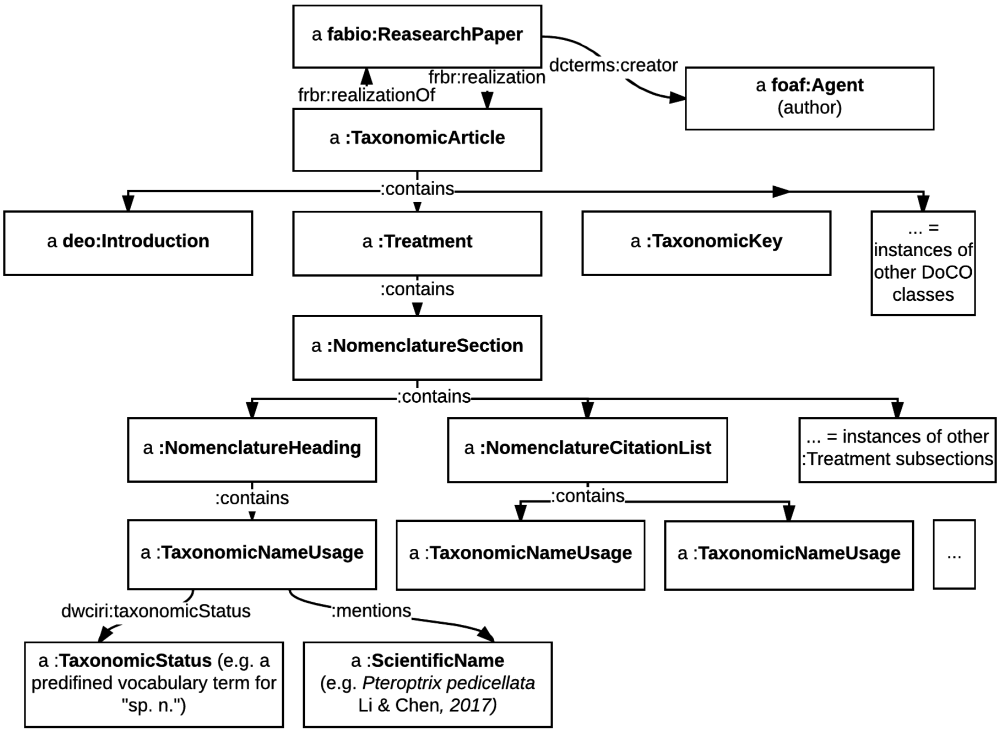


FIGURE 2.1: A graphical representation of the relationships between instances of the publishing-related classes that OpenBiodiv introduces.

Semantics, alignment, and usage

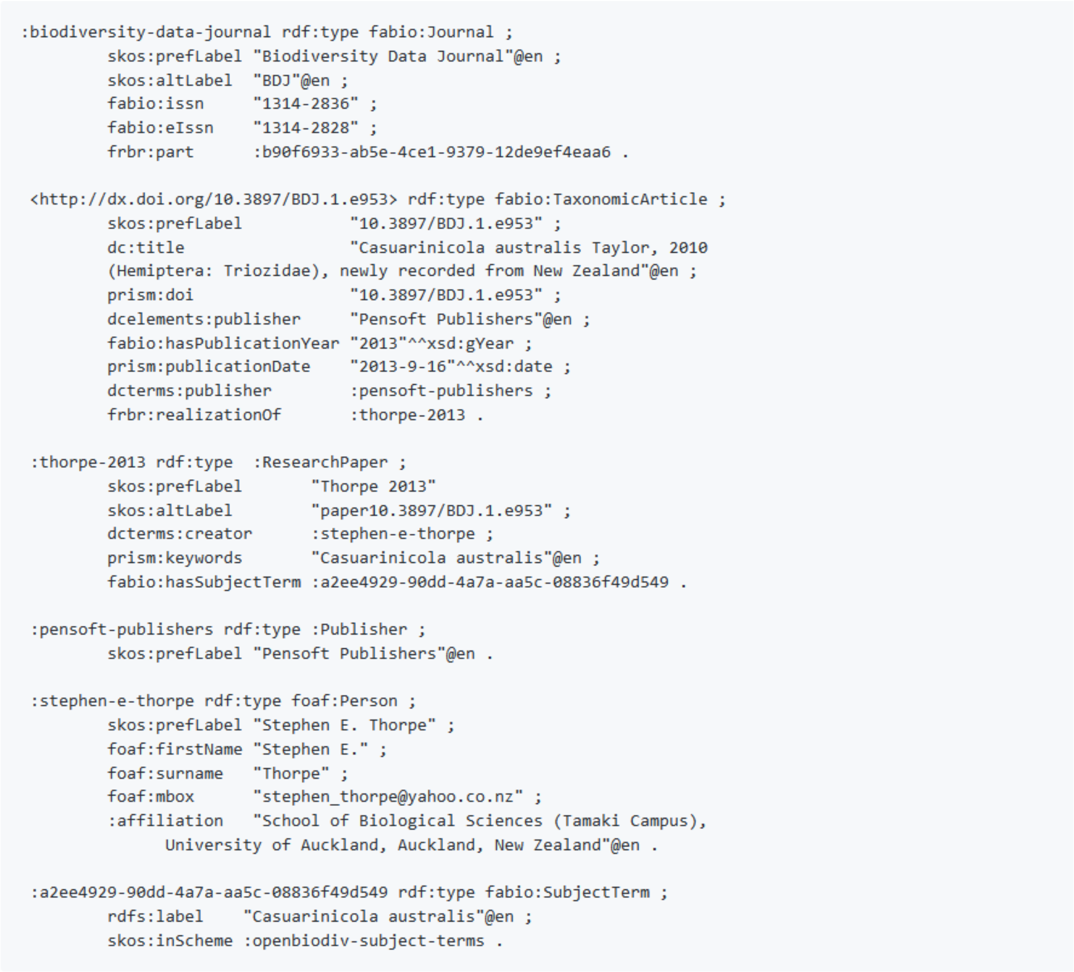
In this section we discuss how the classes and properties that we have introduced align to the Functional Requirements for Bibliographic Records (FRBR) model used by SPAR. In a nutshell taxonomic articles are considered FRBR Expressions of the more abstract FRBR Work that is the intellectual content of the article. Treatments are SPAR discourse elements akin to Introduction, Methods, etc. are also FRBR Expressions. Taxonomic Concepts are their corresponding FRBR Work's.

Figs. 2.2 and 2.3 give example usage in Turtle illustrating these ideas.

2.3.2 Semantic modeling of biological nomenclature

Biological nomenclature is a legacy system with over 200 years of accumulation from before the time of informatics and even from before the time of Darwininan Evolution! It is very hard to model due to complexity and has only partially been covered by the ontologies NOMEN and TNSS (introduced in subsection "Previous work"). With OpenBiodiv-O, I take a bottom-up approach of modeling the use of taxonomic names in articles. Where possible we align OpenBiodiv-O classes to NOMEN.

We have defined the class hierarchy of taxonomic names found in Fig. 2.4. Furthermore, we have introduced the class Taxonomic Name Usage (`:TaxonomicNameUsage`). Taxonomic name usages have been discussed widely in the community (e.g. in Pyle, 2016a); however, the meaning of term remains vague. The abbreviation TNU is used interchangeably for "taxon name usage" and for "taxonomic name usage." In



```
:biodiversity-data-journal rdf:type fabio:Journal ;
        skos:prefLabel "Biodiversity Data Journal"@en ;
        skos:altLabel  "BDJ"@en ;
        fabio:issn     "1314-2836" ;
        fabio:eIssn    "1314-2828" ;
        frbr:part      :b90f6933-ab5e-4ce1-9379-12de9ef4eaa6 .

 <http://dx.doi.org/10.3897/BDJ.1.e953> rdf:type fabio:TaxonomicArticle ;
        skos:prefLabel           "10.3897/BDJ.1.e953" ;
        dc:title                 "Casuarinicola australis Taylor, 2010
        (Hemiptera: Triozidae), newly recorded from New Zealand"@en ;
        prism:doi                "10.3897/BDJ.1.e953" ;
        dcelements:publisher     "Pensoft Publishers"@en ;
        fabio:hasPublicationYear "2013"^^xsd:gYear ;
        prism:publicationDate    "2013-9-16"^^xsd:date ;
        dcterms:publisher        :pensoft-publishers ;
        frbr:realizationOf       :thorpe-2013 .

 :thorpe-2013 rdf:type  :ResearchPaper ;
        skos:prefLabel       "Thorpe 2013"
        skos:altLabel        "paper10.3897/BDJ.1.e953" ;
        dcterms:creator      :stephen-e-thorpe ;
        prism:keywords       "Casuarinicola australis"@en ;
        fabio:hasSubjectTerm :a2ee4929-90dd-4a7a-aa5c-08836f49d549 .

 :pensoft-publishers rdf:type :Publisher ;
        skos:prefLabel "Pensoft Publishers"@en .

 :stephen-e-thorpe rdf:type foaf:Person ;
        skos:prefLabel "Stephen E. Thorpe" ;
        foaf:firstName "Stephen E." ;
        foaf:surname   "Thorpe" ;
        foaf:mbox      "stephen_thorpe@yahoo.co.nz" ;
        :affiliation   "School of Biological Sciences (Tamaki Campus),
              University of Auckland, Auckland, New Zealand"@en .

 :a2ee4929-90dd-4a7a-aa5c-08836f49d549 rdf:type fabio:SubjectTerm ;
        rdfs:label    "Casuarinicola australis"@en ;
        skos:inScheme :openbiodiv-subject-terms .
```

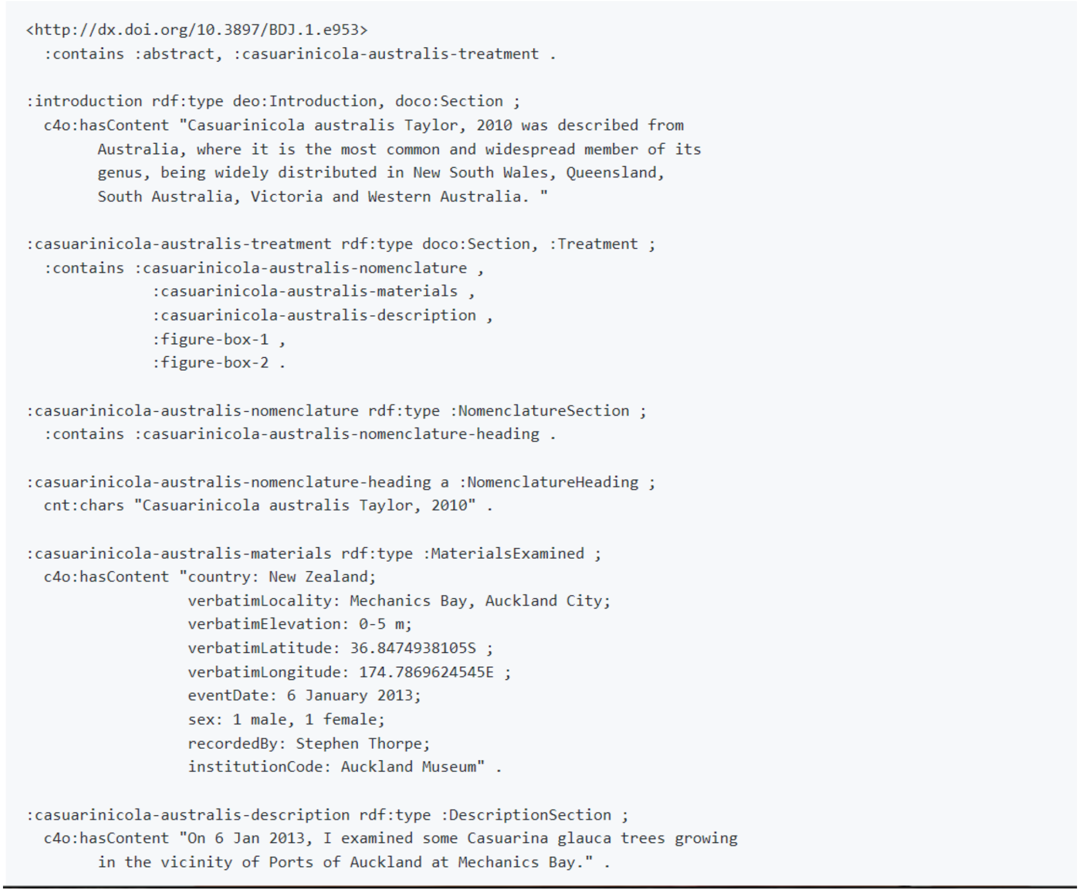
FIGURE 2.2: This example shows how to express the metadata of a taxonomic article with the SPAR Ontologies' model and the classes that OpenBiodiv defines. The code is in Turtle.

OpenBiodiv-O, a taxonomic name usage is the mentioning of a taxonomic name in the text, optionally followed by a taxonomic status.

For example, "*Heser stoevi* Deltschev 2016, sp. n." is a taxonomic name usage. The cursive text followed by the author and year of the original species description is the latinized scientific name. The abbreviation "sp. n." stands for the Latin *species novum*, indicating the discovery of a new taxon.

We also introduce the class Taxonomic Concept Label (`:TaxonomicConceptLabel`). A taxonomic concept label (TCL) is a Linnaean name plus a reference to a publication, where the discussed taxon is circumscribed. The link is via the keyword "sec." (Latin for (*secundum*, Berendsohn, 1995). An example would be "*Andropogon virginicus* var. *tenuispatheus* sec. Blomquist, 1948". Here, Blomquist, 1948 is a valid bibliographic reference to the publication where the concept is circumscribed.

We extracted taxonomic status abbreviations from about 4,000 articles across four taxonomic journals (ZooKeys, Biodiversity Data Journal, PhytoKeys, and MycoKeys) in order to create a taxonomic status vocabulary (see appendices) that covers the eight most common cases (Table 2.2). The Latin abbreviations that have been classified into these classes can be found on the OpenBiodiv-O GitHub page. (See Methods for more details).



```
<http://dx.doi.org/10.3897/BDJ.1.e953>
  :contains :abstract, :casuarinicola-australis-treatment .

:introduction rdf:type deo:Introduction, doco:Section ;
  c4o:hasContent "Casuarinicola australis Taylor, 2010 was described from
        Australia, where it is the most common and widespread member of its
        genus, being widely distributed in New South Wales, Queensland,
        South Australia, Victoria and Western Australia. "

:casuarinicola-australis-treatment rdf:type doco:Section, :Treatment ;
  :contains :casuarinicola-australis-nomenclature ,
              :casuarinicola-australis-materials ,
              :casuarinicola-australis-description ,
              :figure-box-1 ,
              :figure-box-2 .

:casuarinicola-australis-nomenclature rdf:type :NomenclatureSection ;
  :contains :casuarinicola-australis-nomenclature-heading .

:casuarinicola-australis-nomenclature-heading a :NomenclatureHeading ;
  cnt:chars "Casuarinicola australis Taylor, 2010" .

:casuarinicola-australis-materials rdf:type :MaterialsExamined ;
  c4o:hasContent "country: New Zealand;
                  verbatimLocality: Mechanics Bay, Auckland City;
                  verbatimElevation: 0-5 m;
                  verbatimLatitude: 36.8474938105S ;
                  verbatimLongitude: 174.7869624545E ;
                  eventDate: 6 January 2013;
                  sex: 1 male, 1 female;
                  recordedBy: Stephen Thorpe;
                  institutionCode: Auckland Museum" .

:casuarinicola-australis-description rdf:type :DescriptionSection ;
  c4o:hasContent "On 6 Jan 2013, I examined some Casuarina glauca trees growing
        in the vicinity of Ports of Auckland at Mechanics Bay." .
```

FIGURE 2.3: This examples shows how to express the article structure with the help of `:contains`. The code is in Turtle.

TABLE 2.2: OpenBiodiv Taxonomic Status Vocabulary.

Vocabulary Instance QName	Example Abbrev	Comment
`:TaxonomicUncertainty`	*incertae sedis*	Taxonomic Uncertainty
`:TaxonDiscovery`	*sp. n.*	Taxonomic Discovery
`:ReplacementName`	*comb. n.*	Replacement Name
`:UnavailableName`	*nomen dubium*	Unavailable Name
`:AvailableName`	*stat. rev.*	Available Name
`:TypeSpecimenDesignation`	*lectotype designation*	Type Specimen Designation
`:TypeSpeciesDesignation`	*type species*	Type Species Designation
`:NewOccurrenceRecord`	*new country record*	New Occurrence Record (for region)

Based on our analysis of taxonomic statuses, we have identified two Code-compliant patterns of relationship between latinized scientific names (Fig. 2.5). The pattern *replacement name*, implemented via the property `:replacementName`, indicates that a certain Linnaean name should be used instead of another Linnaean name. It covers a wide variety of cases in the Codes, such as, for example, the placement of one species taxon in a new genus ("comb. n."), the correction of a name for nomenclatural reasons ("nomen novum"), or the application of the Principle of Priority for the discovery of synonyms ("syn. nov.", International Commission on Zoological Nomenclature, 2017).

The other pattern is that of *related names* (`:relatedName`). It is a broader pattern, indicating that two names are somehow related. For example, they may be synonyms,

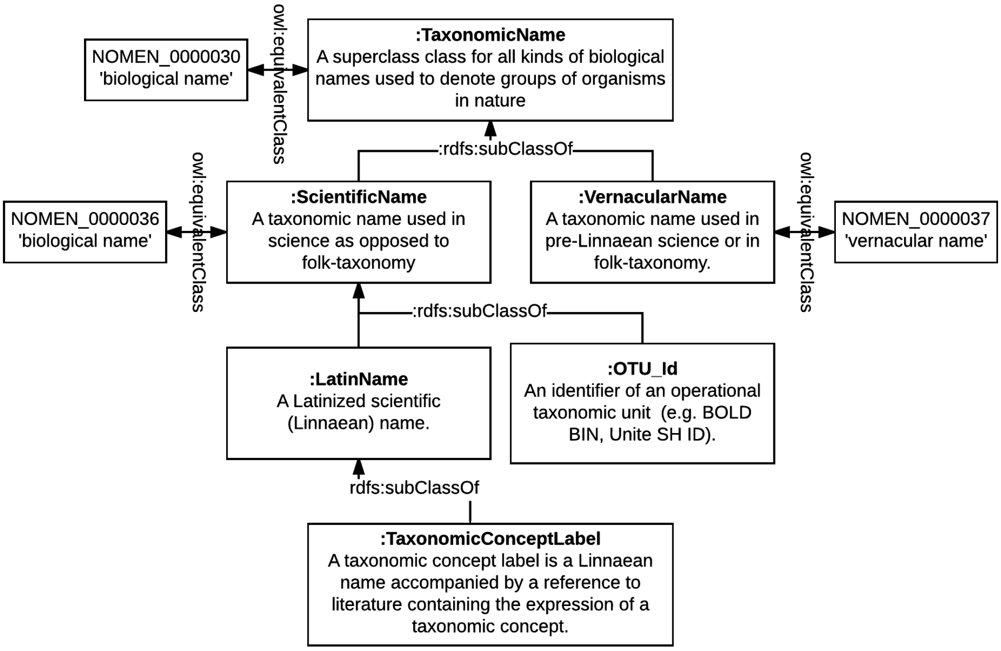

:TaxonomicName
A superclass class for all kinds of biological names used to denote groups of organisms in nature
NOMEN_0000030
'biological name'
owl:equivalentClass
:rdfs:subClassOf
:ScientificName
A taxonomic name used in science as opposed to folk-taxonomy
NOMEN_0000036
'biological name'
owl:equivalentClass
:VernacularName
A taxonomic name used in pre-Linnaean science or in folk-taxonomy.
NOMEN_0000037
'vernacular name'
owl:equivalentClass
:rdfs:subClassOf
:LatinName
A Latinized scientific (Linnaean) name.
:OTU_Id
An identifier of an operational taxonomic unit (e.g. BOLD BIN, Unite SH ID).
rdfs:subClassOf
:TaxonomicConceptLabel
A taxonomic concept label is a Linnaean name accompanied by a reference to literature containing the expression of a taxonomic concept.

Figure 2.4: We created this class hierarchy to accommodate both traditional taxonomic name usages and the usage of taxonomic concept labels and operational taxonomic units.

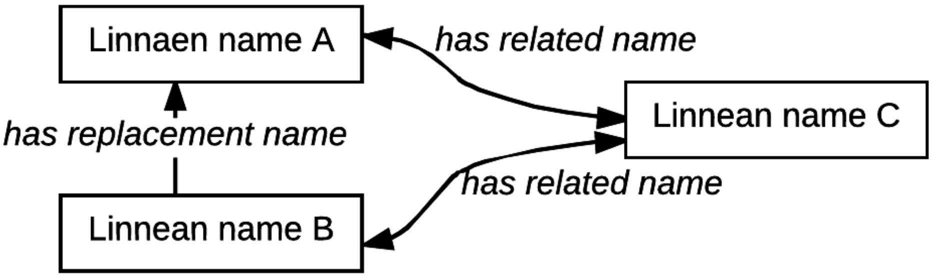


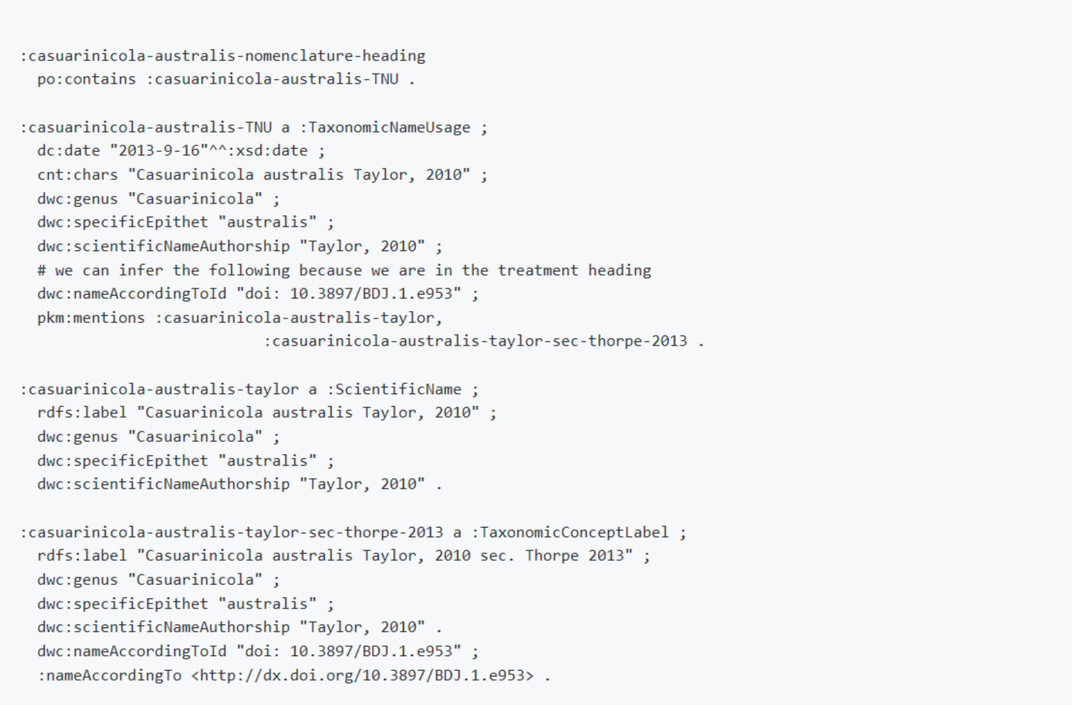
Figure 2.5: Chains of *replacement names* can be followed to find the currently used name. *Related name* indicates that two names are related somehow, but not which one is preferable.

with one replacing the other, or they may point to taxonomically related taxonomic concepts. For example, *Harmonia manillana* (Mulsant, 1866) is related to *Caria manillana* Mulsant, 1866 since, as per Poorani and Booth, 2016, a name-bearing type (lectotype) of *Harmonia manillana* (Mulsant, 1866) sec. Poorani Poorani and Booth, 2016 is named *Caria manillana* Mulsant, 1866.

Semantics, alignment and usage

As evident from Fig. 2.4, OpenBiodiv-O taxonomic names are aligned to NOMEN names.

The linking between text and taxonomic names must pass through the intermediary class Taxonomic Name Usage. As parts of the manuscript, taxonomic name usages link document components to taxonomic names. Taxonomic name usages are *contained* in sections such as Treatment, and *mention* a taxonomic name as illustrated in the example in Fig. 2.6.



```
:casuarinicola-australis-nomenclature-heading
  po:contains :casuarinicola-australis-TNU .

:casuarinicola-australis-TNU a :TaxonomicNameUsage ;
  dc:date "2013-9-16"^^:xsd:date ;
  cnt:chars "Casuarinicola australis Taylor, 2010" ;
  dwc:genus "Casuarinicola" ;
  dwc:specificEpithet "australis" ;
  dwc:scientificNameAuthorship "Taylor, 2010" ;
  # we can infer the following because we are in the treatment heading
  dwc:nameAccordingToId "doi: 10.3897/BDJ.1.e953" ;
  pkm:mentions :casuarinicola-australis-taylor,
                           :casuarinicola-australis-taylor-sec-thorpe-2013 .

:casuarinicola-australis-taylor a :ScientificName ;
  rdfs:label "Casuarinicola australis Taylor, 2010" ;
  dwc:genus "Casuarinicola" ;
  dwc:specificEpithet "australis" ;
  dwc:scientificNameAuthorship "Taylor, 2010" .

:casuarinicola-australis-taylor-sec-thorpe-2013 a :TaxonomicConceptLabel ;
  rdfs:label "Casuarinicola australis Taylor, 2010 sec. Thorpe 2013" ;
  dwc:genus "Casuarinicola" ;
  dwc:specificEpithet "australis" ;
  dwc:scientificNameAuthorship "Taylor, 2010" .
  dwc:nameAccordingToId "doi: 10.3897/BDJ.1.e953" ;
  :nameAccordingTo <http://dx.doi.org/10.3897/BDJ.1.e953> .
```

FIGURE 2.6: This examples shows how taxonomic name usages link document components to taxonomic names. The code is in Turtle.

2.3.3 Semantic Modeling of the Taxonomic Concepts

In OpenBiodiv-O taxonomic names are not the carriers of semantic information about taxa. This task is accomplished by a new class, Taxonomic Concept (`:TaxonomicConcept`). A taxonomic concept is the theory that a taxonomist forms about a taxon in a scholarly biological taxonomic publication and thus always has a taxonomic concept label. We also introduce a more general class, Operational Taxonomic Unit (`:OperationalTaxonomicUnit`) that can be used for all kinds of taxonomic hypotheses, including ones that don't have a proper taxonomic concept label. The class hierarchy has been illustrated in Fig. 2.7.

Taxonomic concepts are related to taxonomic names—including taxonomic concept labels—via the property *has taxonomic name* (`:taxonomicName`) and its sub-properties mimicking in their range the hierarchy of taxonomic names that we introduced earlier. We have defined a property specifically to link taxonomic concepts to taxonomic concept labels, *has taxonomic concept label* (`:taxonomicConceptLabel`). The property hierarchy diagram is shown in Fig. 2.8.

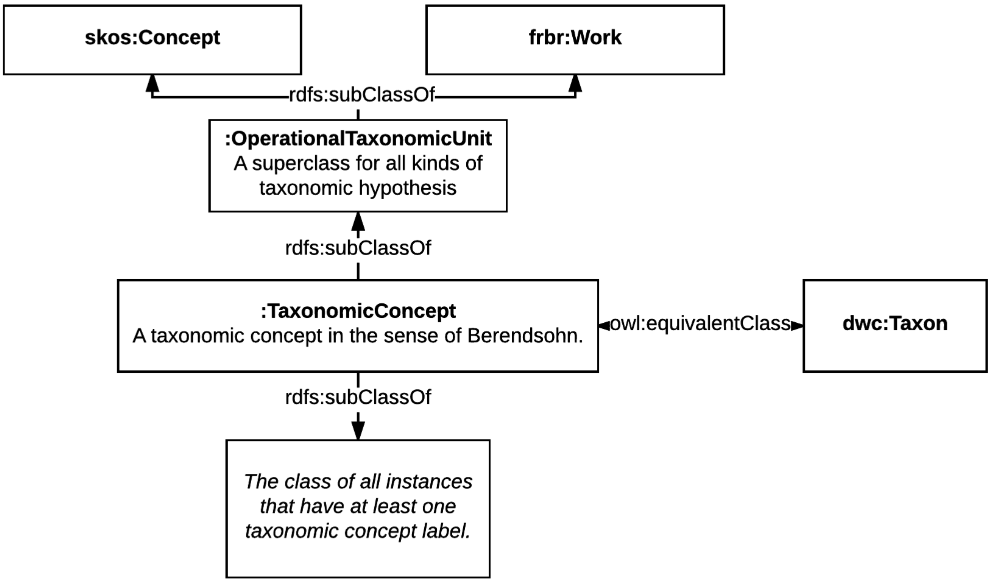

skos:Concept
frbr:Work
rdfs:subClassOf
:OperationalTaxonomicUnit
A superclass for all kinds of taxonomic hypothesis
rdfs:subClassOf
:TaxonomicConcept
A taxonomic concept in the sense of Berendsohn.
owl:equivalentClass
dwc:Taxon
rdfs:subClassOf
The class of all instances that have at least one taxonomic concept label.

FIGURE 2.7: A taxonomic concept is a `skos:Concept`, a `frbr:Work`, a `dwc:Taxon` and has at least one taxonomic concept label.

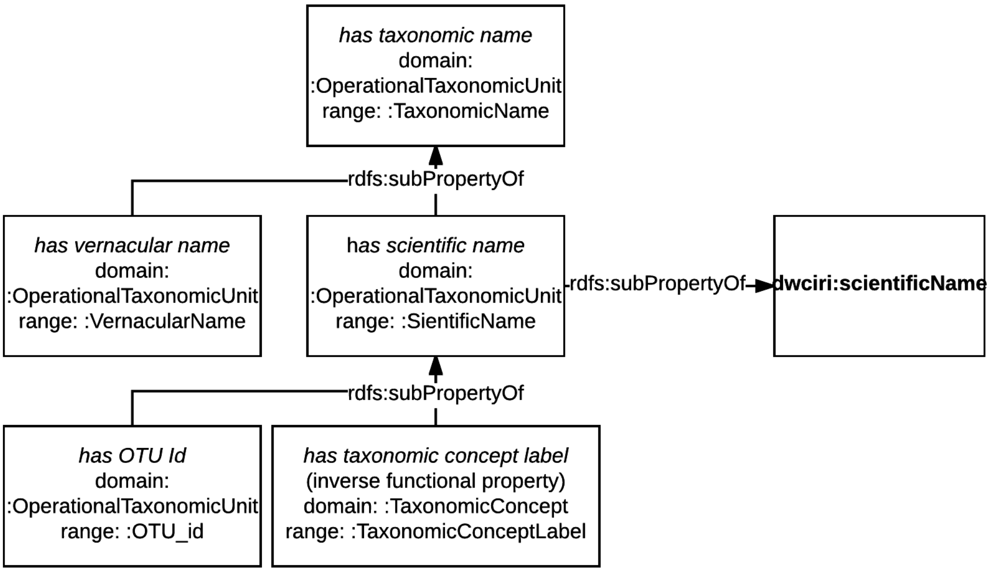

has taxonomic name
domain: :OperationalTaxonomicUnit
range: :TaxonomicName
rdfs:subPropertyOf
has vernacular name
domain: :OperationalTaxonomicUnit
range: :VernacularName
has scientific name
domain: :OperationalTaxonomicUnit
range: :SientificName
rdfs:subPropertyOf
dwciri:scientificName
rdfs:subPropertyOf
has OTU Id
domain: :OperationalTaxonomicUnit
range: :OTU_id
has taxonomic concept label
(inverse functional property)
domain: :TaxonomicConcept
range: :TaxonomicConceptLabel

FIGURE 2.8: Property hierarchy is aligned with the taxonomic name class hierarchy and with DarwinCore.

There are two ways to relate taxonomic concepts to each other (Fig. 2.9). As we pointed out earlier, historically taxonomic concepts form the hierarchy known as biological taxonomy. To express such simple semantic relations, it is fully sufficient to use the SKOS semantic vocabulary Miles and Bechofer.

However, these simple relationships are not well suited for machine reasoning.

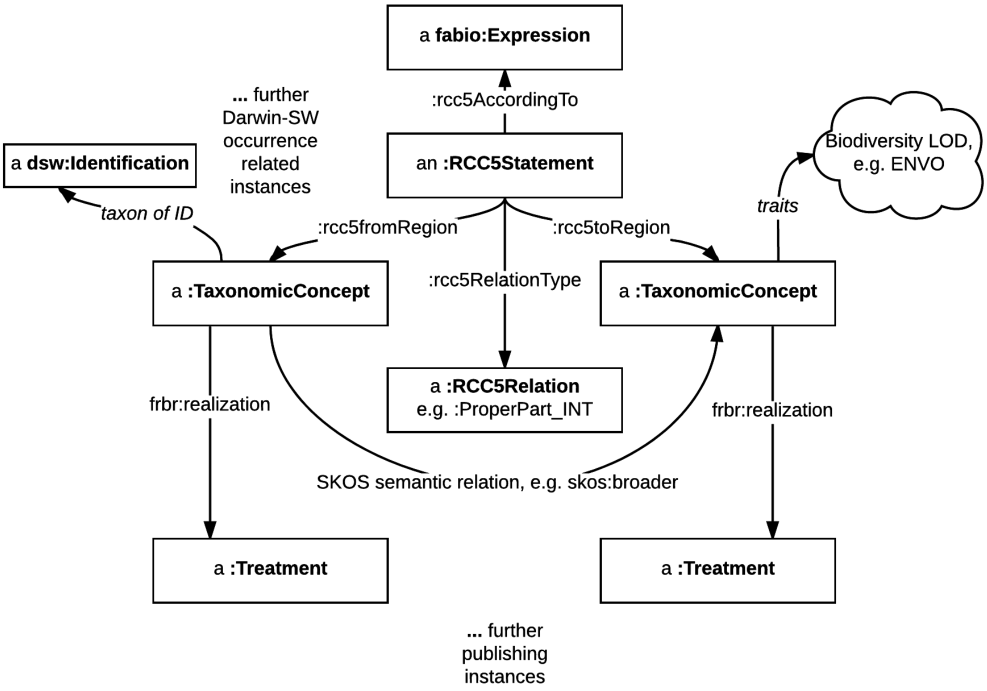


FIGURE 2.9: In order to express an RCC-5 relationship between concepts, create an `:RCC5Sgtatement` and use the corresponding properties to link two taxonomic concepts via it. Further, taxonomic concepts are linked to traits (e.g. ecology in ENVO), occurrences (e.g. Darwin-SW) and realize treatments.

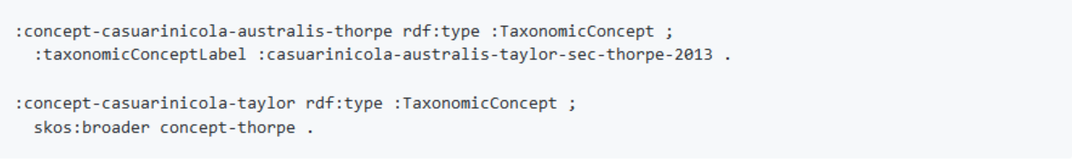
This is why Franz and Peet Franz and Peet, 2009 suggested, building on previous work by e.g. Koperski et al., 2000, to use the RCC-5 language to express relationships between taxonomic concepts. Furthermore, the Euler (Chen et al., 2014) program was developed, which uses Answer Set Programming (ASP) to reason over RCC-5 taxonomic relationships. An answer set reasoner is not part of OpenBiodiv as this task can be accomplished by Euler; however, we have provided an RCC-5 dictionary class (`:RCC5Dictionary`), an RCC-5 relation term class (`:RCC5Relation`), a vocabulary of such terms to express the RCC-5 relationships in RDF (see appendices), as well as a class and properties to express RCC-5 statements (`:RCC5Statement`, `:rcc5Property`, and subproperties).

Semantics and alignment

In this section taxonomic concepts are aligned to DarwinCore (DwC) and a discussion of how taxonomic concepts related to each either via simple relations (SKOS) and fine-grained (RCC-5) is presented. Also the relationships between biological names and scieintific concepts are discussed. We treat instances of our class Taxonomic Concept as functionally equivalent to DwC Taxa. We can now list what types of relationships between names and taxonomic concepts are allowed: (1) The relationship between a taxonomic concept and a name that is not a taxonomic concept label is many-to-many—i.e. one Linnaean name can be a mention of multiple taxonomic concepts, and one taxonomic concept may have multiple Linnaean names. (2) The relationship between a taxonomic concept and a taxonomic concept label is one-to-many: while a taxonomic concept may have more than one (at least one is needed) labels, every label uniquely identifies a concept. These logical restrictions make taxonomic concept labels into unique identifiers to taxonomic concepts, something that Linnaean names are not.

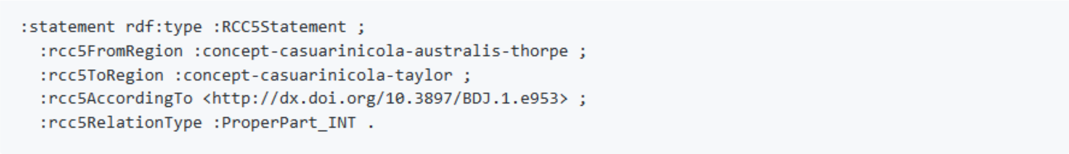
Usage

In Fig. 2.10, Fig. 2.11, and Fig. 2.12, and Fig 2.13) we provide some useful examples.



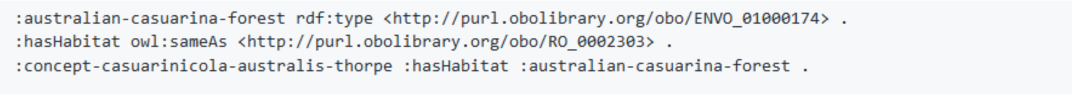
```
:concept-casuarinicola-australis-thorpe rdf:type :TaxonomicConcept ;
  :taxonomicConceptLabel :casuarinicola-australis-taylor-sec-thorpe-2013 .

:concept-casuarinicola-taylor rdf:type :TaxonomicConcept ;
  skos:broader concept-thorpe .
```

FIGURE 2.10: We can use SKOS semantic properties to illustrate simple relationships between taxonomic concepts.



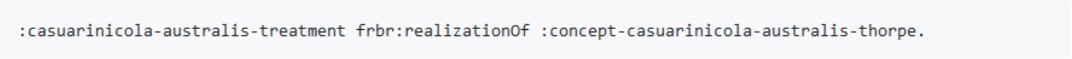
```
:statement rdf:type :RCC5Statement ;
  :rcc5FromRegion :concept-casuarinicola-australis-thorpe ;
  :rcc5ToRegion :concept-casuarinicola-taylor ;
  :rcc5AccordingTo <http://dx.doi.org/10.3897/BDJ.1.e953> ;
  :rcc5RelationType :ProperPart_INT .
```

FIGURE 2.11: In order to express an RCC-5 relationship between concepts, create an `:RCC5Sgtatement` and use the corresponding properties to link two taxonomic concepts via it. SKOS relations relate concepts directly.



```
:australian-casuarina-forest rdf:type <http://purl.obolibrary.org/obo/ENVO_01000174> .
:hasHabitat owl:sameAs <http://purl.obolibrary.org/obo/RO_0002303> .
:concept-casuarinicola-australis-thorpe :hasHabitat :australian-casuarina-forest .
```

FIGURE 2.12: We create a shortcut for *has habitat* and instance of the "forest biome" and link them to our taxonomic concept in order to express the fact that specimens of it have been found to live in *Casuarina* trees.



```
:casuarinicola-australis-treatment frbr:realizationOf :concept-casuarinicola-australis-thorpe.
```

FIGURE 2.13: A treatment is the realization of a taxonomic concept.

2.4 Discussion

OpenBiodiv-O is—together with the Treatment Ontologies (Catapano and Morris, 2016)—the first effort to model taxonomic articles as RDF. It introduces classes and properties in the domains of biodiversity publishing and biological taxonomy and aligns them with the SPAR Ontologies, the Treatment Ontologies, the Open Biomedical Ontologies (OBO), TaxPub, NOMEN, and DarwinCore. We believe this introduction bridges the ontological gap that we had outlined in our aims and allows for the creation of a Linked Open Dataset (LOD) of biodiversity information (biodiversity knowledge graph, Senderov and Penev, 2016; Page, 2016).

Furthermore, this biodiversity knowledge graph, together with this ontology, additional semantic rules, and user software forms the OpenBiodiv system. OpenBiodiv, as any taxonomic information system should, has taxonomic names as a key building block. For any given taxonomic name, the user will be able to rely on two patterns—*replacement name* and *related name*—to get answers to two questions of high importance to the working taxonomist. First: what is the current and historical usage of any given Linnaean name? Second: given a particular name, what other related names ought to be considered in a taxonomic discussion?

In this section we carry out a discussion how the model of OpenBiodiv can be used to store *multiplicity of opinion* about taxonomic relationships and thus democratize the taxonomic process. We further discuss the usefulness of OpenBiodiv to *answer competency questions* from biological taxonomy. These will be touched upon more in the next chapter.

2.5 Conclusions

The chapter provides an informal conceptualization of the taxonomic process and a formalization in OpenBiodiv-O. It introduces classes and properties in the domains of biodiversity publishing and biological systematics and aligns them with the important domain-specific ontologies. By bridging the ontological gap between the publishing and the biodiversity domains, it will enable the creation of Open Biodiversity Knowledge Management System, consisting of (1) the ontology itself; (2) a Linked Open Dataset (LOD) of biodiversity information (biodiversity knowledge graph); and (3) user interface components aimed at searching, browsing and discovering knowledge in big corpora of previously dispersed scholarly publications. Through the usage of taxonomic concepts, we have included mechanisms for democratization of the scholarly process and not forcing a taxonomic opinion on the users.

Chapter 3

Summary of Chapter 3: OpenBiodiv Linked Open Dataset

In Chapter 3 I explore in detail the data sources and their data models.

I, with the help of my support team—see the Acknowledgements in the back—have created a Linked Open Dataset, OpenBiodiv LOD, comprising biodiversity information extracted from Pensoft journals and from Plazi Treatment Bank, and which was integrated with the GBIF Taxonomic Backbone. As ontology, I use the new OpenBiodiv-O developed through the course of the dissertation. I propose to the biodiversity informatics community to use OpenBiodiv LOD as the central point for a biodiversity knowledge graph. OpenBiodiv LOD is an RDF dataset adhering to the principles of Linked Open Data. It is available under `http://graph.openbiodiv.net`, which provides a SPARQL endpoint for it.

OpenBiodiv LOD is a synthetic dataset. It does not contain previously unpublished data. Instead it integrates information previously found in academic journals and databases into one dataset. It also contains extracted, previously inaccessible information from the original datasets in the form of relations. In the next few paragraphs we discuss the sources of information that were combined to from OpenBiodiv LOD and the types of resources that have been extracted, as well as the overall data model. We also discuss the principles of Linked Open Data that tie everything together. The chapter ends with many examples of queries on the dataset and with a technical discussion of how it was generated.

3.1 Data Sources

The data in OpenBiodiv at the time of writing this thesis comes from three major sources: the GBIF Backbone Taxonomy (GBIF Secretariat, 2017b), journal articles published by Pensoft, and Plazi Treatment Bank (Fig. 3.1).

3.1.1 GBIF Backbone Taxonomy

GBIF is the largest international repository of occurrence data, i.e. data about the presence of an organism of a given taxon at a given place and time. GBIF allows its users to do searches on its occurrence data utilizing a taxonomic hierarchy. For example, it is possible to query the database for occurrences of organisms belonging to a specific genus: a search for the beetle genus *Harmonia* sec. GBIF Secretariat, 2017b on 30 June 2018 returned 575,376 results. This search is possible thanks to the GBIF Backbone Taxonomy also known as Nub (GBIF Secretariat, 2017b). Nub is a database organizing taxonomic concepts in a hierarchy covering all names used in occurrence records harvested by GBIF. It is a single synthetic (algorithmically generated) management classification with the goal of covering all names present in

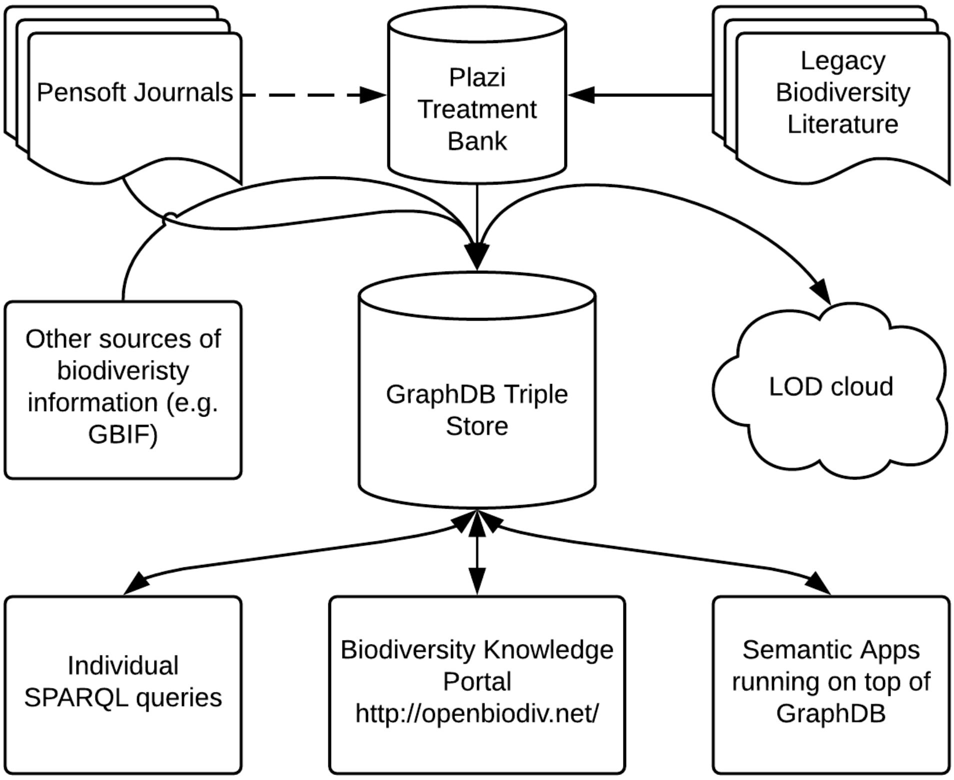

Pensoft Journals
Plazi Treatment Bank
Legacy Biodiversity Literature
Other sources of biodiveristy information (e.g. GBIF)
GraphDB Triple Store
LOD cloud
Individual SPARQL queries
Biodiversity Knowledge Portal http://openbiodiv.net/
Semantic Apps running on top of GraphDB

Figure 3.1: A simplified version of the OpenBiodiv architecture presented in Chapter 1 focusing on the sources of information.

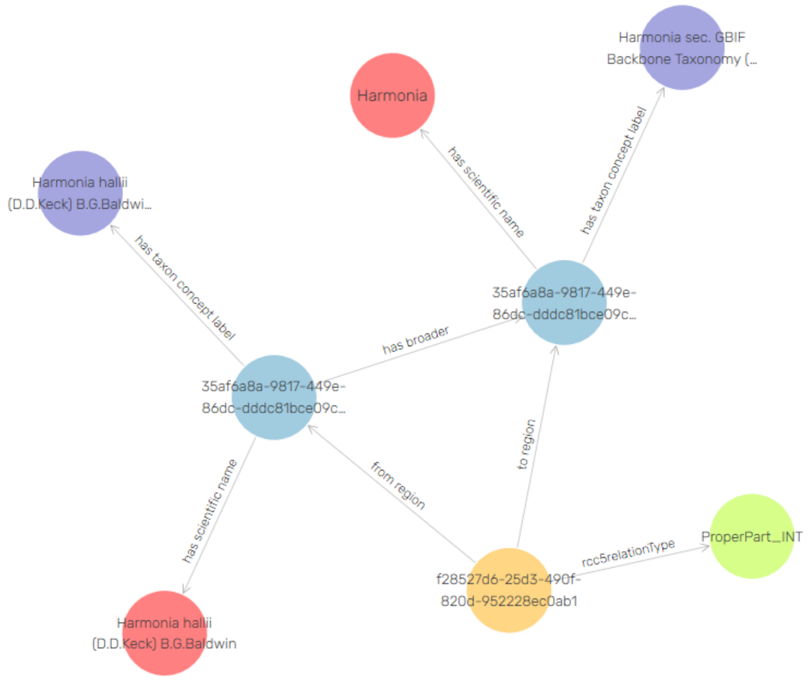

Harmonia sec. GBIF Backbone Taxonomy (...
Harmonia
Harmonia halii (D.D.Keck) B.G.Baldwi...
has scientific name
has taxon concept label
has taxon concept label
35af6a8a-9817-449e-86dc-dddc81bce09c...
has broader
35af6a8a-9817-449e-86dc-dddc81bce09c...
to region
from region
has scientific name
ProperPart_INT
rcc5relationType
f28527d6-25d3-490f-820d-952228ec0ab1
Harmonia halii (D.D.Keck) B.G.Baldwin

FIGURE 3.2: Illustration of the representation of hierarchical information imported from the GBIF Backbone Taxonomy as two taxonomic concepts, *Harmonia halii* sec. GBIF Secretariat, 2017a and *Harmonia* sec. GBIF Secretariat, 2017a. Each concept has an associated scientific name via *has scientific name*; however, the hierarchical information is not encoded in the names. The hierarchical relationship between *Harmonia halii* sec. GBIF Secretariat, 2017a and *Harmonia* sec. GBIF Secretariat, 2017a is encoded both as SKOS *has broader* and reified via the RCC-5 relationship encoded in `f28527d6-25d3-490f-820d-952228ec0ab1`.

GBIF's datasets. Thus, the GBIF backbone does not represent an expert consensus on how taxa are hierarchically arranged according to evolutionary criteria in Nature.

Keeping in mind this critique, it is evident how the backbone taxonomy allows GBIF to integrate name based information from diverse sources such as Encyclopedia of Life (EOL), Genbank, or the International Union for Conservation of Nature, and provides a facility for taxonomic searching and browsing.

In order to grant the same capabilities to OpenBiodiv, we have imported Nub as instances of `openbiodiv:TaxonomicConcept` according to OpenBiodiv-O (Fig. 3.2).

3.1.2 Pensoft and Plazi

All valid articles from the journals published by Pensoft listed in Table 3.1 have been converted to RDF and stored in the biodiversity knowledge graph. Additionally, all valid taxonomic treatments from Plazi Treatment Bank have been converted to RDF and stored in the graph as well. Furthermore, the RDF-ization procedure is triggered automatically on a weekly basis and thus the semantic database is always updated with the newest articles published by Pensoft and newest taxonomic treatments extracted by Plazi. The RDF-ization is made possible by the fact that all Pensoft journals are published as XML according to TaxPub, an extension of the NLM/NCBI journal publishing DTD for taxonomic description (Catapano, 2010) and, similarly, all Plazi treatments follow the TaxonX XML Schema (Penev et al., 2011) (Fig. 3.3).

LISTING 3.1: Taxonomic name usage of the name *P. emarginaticeps* in Taxpub. Name parts are tagged with `tp:taxon-name-part` and the expansion of abbreviations (regularization) is marked up with the attribute `reg`

```
<tp:taxon-name>
  <tp:taxon-name-part taxon-name-part-type="genus" reg="Pristaulacus">
    P.
  </tp:taxon-name-part>
  <tp:taxon-name-part taxon-name-part-type="species" reg="emarginaticeps">
    emarginaticeps
  </tp:taxon-name-part>
  <tp:taxon-name-part taxon-name-part-type="authority">
    Turner 1922
  </tp:taxon-name-part>
</tp:taxon-name>
```

TABLE 3.1: RDF-ized biodiversity journals published by Pensoft.

Journal Name	Submission Style	Number of Articles
ZooKeys	Word document	3829
PhytoKeys	Word document	537
MycoKeys	Word document	127
Biodiversity Data Journal	Web based (ARPHA)	490
Journal of Orthoptera Research	Word document	32

TABLE 3.2: Datatypes marked up in TaxPub and TaxonX articles and the corresponding RDF types of the generated RDF resources. The TaxPub and TaxonX columns contain boolean values indicating whether the information about the datatype is retrieved from files encoded in the corresponding schema.

Datatype	TaxPub	TaxonX	RDF Type
Article metadata	T	T	`fabio:JournalArticle` and related
Keyword group	T	F	`openbiodiv:KeywordGroup`
Abstract	T	T	`sro:Abstract`
Title	T	F	`doco:Title`
Author	T	T	`foaf:Person`
Introduction section	T	F	`deo:Introduction`
Discussion section	T	T	`orb:Discussion`
Treatment section	T	T	`openbiodiv:Treatment`
Nomenclature section	T	T	`openbiodiv:NomenclatureSection`
Materials examined	T	T	`openbiodiv:MaterialsExamined`
Diagnosis section	T	T	`openbiodiv:DiagnosisSection`
Distribution section	T	T	`openbiodiv:DistributionSection`
Taxonomic key	T	T	`openbiodiv:TaxonomicKey`
Figure	T	T	`doco:Figure`
Taxonomic name usage	T	T	`openbiodiv:TaxonomicNameUsage`

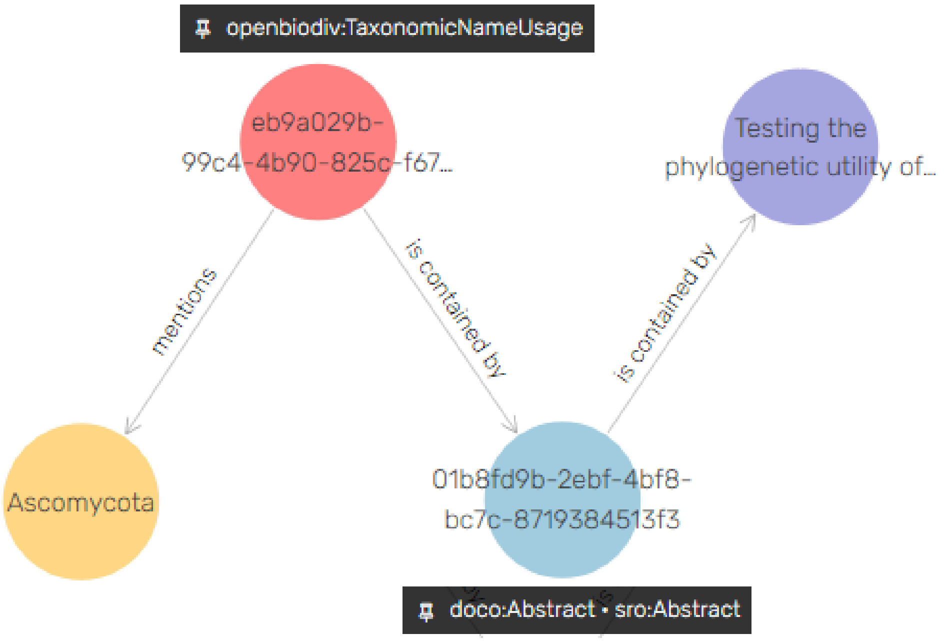

openbiodiv:TaxonomicNameUsage
eb9a029b-99c4-4b90-825c-f67...
Testing the phylogenetic utility of...
mentions
is contained by
is contained by
Ascomycota
01b8fd9b-2ebf-4bf8-bc7c-8719384513f3
doco:Abstract • sro:Abstract

Figure 3.3: The taxonomic name usage (`openbiodiv:eb9a029b-99c4-4b90-825c-f670fb88900d`) is linked to the scientific name it mentions, *Ascomycota* and to the part of the article (abstract) that it is contained in.

3.2 Linked Open Data

Linked Open Data (LOD, Heath and Bizer, 2011) is a concept of the Semantic Web (Berners-Lee et al., 2001) applied to ensure that data published on the Web is reusable, discoverable and most importantly to ensure that pieces of data published by different entities can work together. The principles of LOD are the following (Heath and Bizer, 2011)

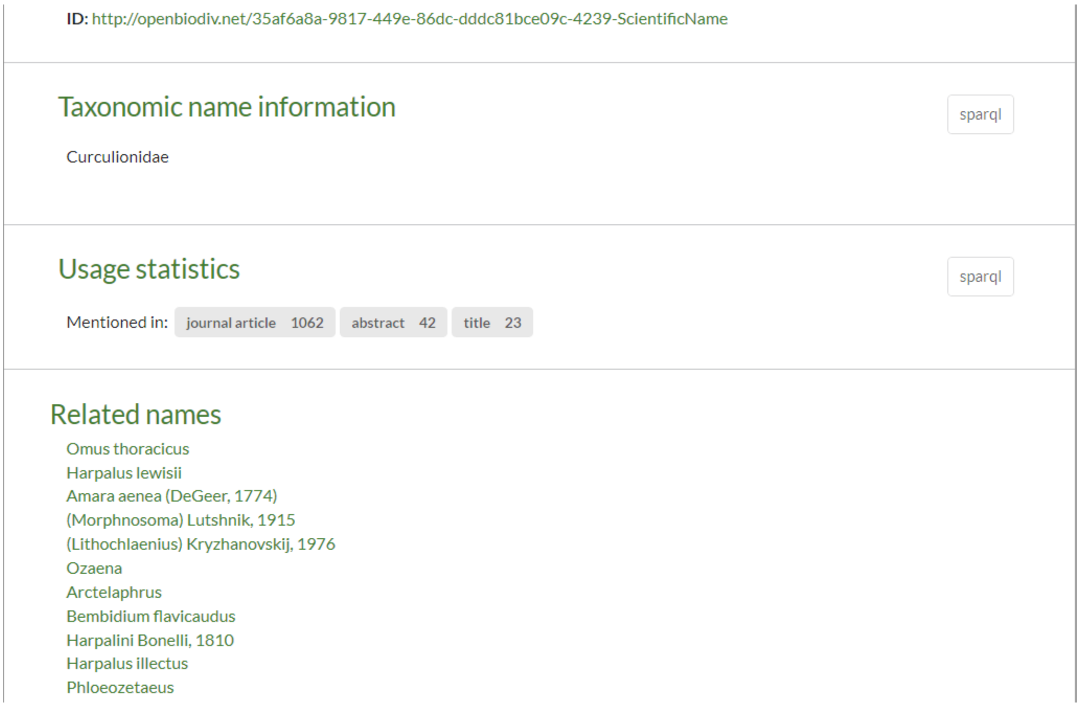
1. Use URIs as names for things.
2. Use HTTP URIs so people can lookup these things.
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
4. Include links to other URIs so they can discover more things.

We have followed these guidelines when creating the OpenBiodiv LOD. We will now discuss each of these points separately.

3.2.1 Usage of URI's as resource identifiers

Every instance in OpenBiodiv LOD is uniquely identifiable by a HTTP URI of the following form: `http://openbiodiv.net/uuid-(suffix)`. All instance identifiers in OpenBiodiv LOD follow this schema. The optional suffix field is assigned only to resources extracted from GBIF.

In this subsection we further discuss how identifiers are assigned to resources extracted from Pensoft and Plazi as well as to the GBIF taxonomic concepts.



ID: http://openbiodiv.net/35af6a8a-9817-449e-86dc-dddc81bce09c-4239-ScientificName

Taxonomic name information

Curculionidae

Usage statistics

Mentioned in: journal article 1062 abstract 42 title 23

Related names

Omus thoracicus
Harpalus lewisii
Amara aenea (DeGeer, 1774)
(Morphnosoma) Lutshnik, 1915
(Lithochlaenius) Kryzhanovskij, 1976
Ozaena
Arctelaphrus
Bembidium flavicaudus
Harpalini Bonelli, 1810
Harpalus illectus
Phloeozetaeus

FIGURE 3.4: Visualization.

3.2.2 Usage of HTTP URI's and dereferencing

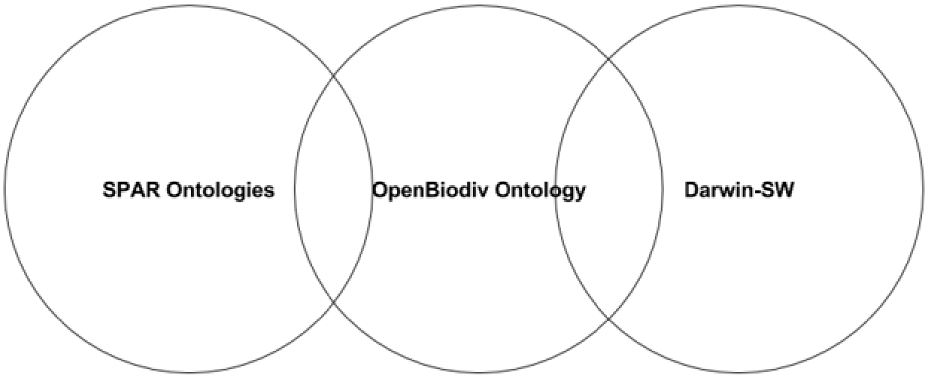
As per the Linked Data Principles, we use dereferenceable HTTP URIs for our resources. For example, if a web-browser opens http://openbiodiv.net/35af6a8a-9817-449e-86dc-dddc81bce09c-4239-ScientificName a web-page is displayed (Fig. 3.4) providing useful information for the name such as where it used and other names are related to it. Also it is possible to request OpenBiodiv resources via Curl with the header `Content-Type: application/rdf+xml` and an RDF representation of the resources is returned.

3.2.3 Linking to other resources

First, all resources in OpenBiodiv form a graph (there are no disconnected parts). The data model is discussed in the next section. Second, taxonomic names are linked to external databases via `dwc:taxonID`. These are strings containing GBIF ID's, ZooBank ID's, LSID's, etc. Unfortunately as HTTP URI's have not gained popularity in the biodiversity informatics community, the only true resource-id-to-resource-id links are within OpenBiodiv itself. However, we hope that the introduction of OpenBiodiv LOD contributes to the amelioration of this situation.

3.3 Data Model

When creating the RDF graph we have conformed to the OpenBiodiv Ontology described in Chapter 2 and well-established community ontologies (Fig. 3.5). In particular, (1) we use the Semantic Publishing and Referencing Ontologies (SPAR, Peroni, 2014) to model entities from publishing such as Journal, Article, Section, Figure, Table, and so on; and (2) we use the DarwinCore (DwC, Wieczorek et al., 2012)



Work	Treatment	Identificiation
Expression	Nomenclature	Organism
Journal	Nomenclature Heading	Token
Journal Article	Taxonomic Name	Occurrence
Research Paper	Latin Name	Event
FrontMatter	Taxonomic Concept Label	Agent
Body Matter	Taxonomic Concept	Location
Back Matter	Taxonomic Status	
Section	Vocabulary	
Table	RCC5 Vocabulary	
Figure		

FIGURE 3.5: OpenBiodiv-O is an ontology that links the publishing domain with the biodiversity domain. Major resource types covered by each of the ontology families are given in the box below the Venn diagram. Important resources from the publishing domain are listed in the leftmost column and from biodiversity informatics in the rightmost column. The middle one covers important OpenBiodiv-O resources.

community standard and its extension, the Darwin-SW (Baskauf and Webb, 2016) ontology, to model entities the biodiversity domain.

SPAR provides facilities to deal with the dichotomy between the abstract representation of knowledge through the class Work and its concrete representation through the class Expression. For example, a `fabio:JournalArticle` can be the realization of a `fabio:ResearchPaper`. On the other hand, the DwC community standard gives a standard way to express properties from taxonomy and biodiversity science and its extension Darwin-SW a way to reify elements of an occurrence instance such as Identification, Organism, Token, and so on. A caveat: the current version of OpenBiodiv-LOD does not store yet occurrence information but all necessary infrastructure is in place to include them in the next release.

3.4 Examples of SPARQL queries

As SPAR, DwC, and OpenBiodiv-O have already been explained elsewhere, we shall illustrate the data model by issuing sample SPARQL queries illuminating aspects of it.

3.4.1 Simple queries

In this section, we give some simple queries. For example, how to search for an author, for a scientific name, etc.

Query the article structure

A unique feature of OpenBiodiv LOD is that articles are broken down into their components (see e.g. Table 3.2 later in this Chapter) and mentions (e.g. taxonomic name usages) connected to the specific part of the article and not just to the article in general. We illustrate how to build queries utilizing this structure.

Query for taxonomic concepts

A key feature of OpenBiodiv-O is that it allows for the separation of taxonomic concepts from scientific names. Scientific names are linked both to the components of an article that mentions them and to taxonomic concepts. To illustrate this, we can create a query uniting information from concepts from the GBIF Backbone Taxonomy with semantics coming from the article structure.

Fuzzy Queries via Lucene

The SPARQL endpoint of OpenBiodiv LOD supports fuzzy matching via a Lucene connector (Ontotext, 2018). In taxonomy, this can be a very useful as due to multiplicity of taxonomic names and the complexities of Latin grammar, one often does not remember the correct spelling of a name. This can lead to no matches in an exact search even though the system may contain information about that name. We illustrate how to do Lucene queries in OpenBiodiv via SPARQL.

3.4.2 Competency question answering via SPARQL

At the end of Chapter 2 I suggested some competency questions that may be answered by OpenBiodiv. In this subsection I show how these can be answered with the help of OpenBiodiv.

Validity of a taxonomic name

Of central importance is the question of whether a given taxonomic name is valid or not. We give the formal criteria on judging the validity of a taxonomic name and translate these into SPARQL.

Investigation of the impact of the lost collections of Museu Nacional

We conclude the discussion of SPARQL queries by showing how OpenBiodiv can be used to assess the impact of the tragically lost collection of the Museu Nacional de Rio de Janeiro (MNRJ).

3.5 Dataset Generation

In the previous section on sources we examined the data formats that each source provides. The inputs are either XML (Pensoft and Plazi) or CSV (GBIF). Thus, the raw data-streams are semi-structured and the dataset generation problem can be thought of as an information retrieval and transformation problem. The input is encoded in three different data models—DarwinCore CSV (GBIF), TaxPub XML (Pensoft), and TaxonX XML (Plazi). The output of the transformation pipeline is knowledge represented in a fully-structured way according to the ontology.

3.5.1 Obtaining the data

The first step before running any transformation is to obtain the raw inputs. GBIF's taxonomic backbone is available under `<https://www.gbif.org/dataset/d7dddbf4-2cf0-4f39-9b2a-bb099caae36c>`. There is an RSS feed from which Plazi's treatments can be downloaded on a daily basis under `<http://tb.plazi.org/GgServer/xml.rss.xml>`. Each of Pensoft's journals has a public API endpoint under `<http://[journal_name].pensoft.net/lib/journal_archive.php>`, where `[journal_name]` ought to be replaced with the name of the Pensoft journal. E.g. `bdj` to make `<http://bdj.pensoft.net/lib/journal_archive.php>`.

3.5.2 Tools

In order to carry out the dataset generation we made use of the following tools:

1. RDF4R R package[1], which is described in Chapter 4 and deals with all RDF-related issues such as accessing a triple store, serializing the in-memory resource representations to Turtle files, etc.

2. ROpenBio R package[2], which implements the data retrieval and transformations described in this chapter.

3. TSV4RDF, which is a PHP library for mapping CSV to RDF developed by Pensoft. It is closed-source and developed outside of the scope of the dissertation and is not discussed in detail.

4. The OpenBiodiv base[3], which contains scripts needed for the initialization and updating of the database.

In the rest of the section we describe the transformation from XML as it is implemented in ROpenBio. We do not describe the TSV4RDF transformation of GBIF to RDF as it is a closed source product.

3.5.3 XML to RDF transformation

In order to transform an article represented as an XML document to RDF, we make use of the hierarchical nature of XML and solve the problem recursively with the following Extractor procedure in Algorithm 1. The extractor's procedure input is an XML node and its output is the RDF corresponding to the XML node. The extractor procedure has three essential steps: atoms extraction, RDF constructions from the extracted atoms, a divide-and-conquer step that recursively calls itself and unites the results. Extraction of a whole article is achieved by calling the Extractor on the root node of the article.

[1] RDF4R package on GitHub `<https://github.com/vsenderov/rdf4r>`

[2] ROpenBio R package on GitHub `<https://github.com/pensoft/ropenbio>`

[3] OpenBiodiv Base `<https://github.com/vsenderov/OpenBiodiv>`

Algorithm 1 The Extractor procedure

```
procedure EXTRACTOR(XML Node $X$)
    $a \leftarrow$ extract atoms of $X$                      ▷ Atoms extraction
    $r \leftarrow$ construct RDF from $a$                    ▷ RDF construction
    $C \leftarrow$ find relevant sub-nodes of $X$            ▷ Recursively applies itself
    $R \leftarrow$ apply Extractor on each $C_i \in C$
    return $r \bigcup R$
end procedure
```

Atoms extraction

In this subsection we elaborate on the text-fields of the XML (atoms) are extracted in our framework utilizing the XPATH query language.

RDF Generation

Once the atoms have been extracted they can be put together as RDF. Conceptually, this is straightforward as for each atom we know its type and therefore we know which RDF property to use. The author example is given in Listing 3.2.

LISTING 3.2: RDF snippet of an author. This is a somewhat idealized situation in which the language of the address was available from the article.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

:a a foaf:Person ;
   rdfs:label "Aijaz␣Ahmad␣Wachkoo".
   :affiliation "Central␣Institute␣of␣Temperate␣Horticulture,␣Srinagar,␣Jammu␣&␣Kashmir,␣India"@en ;
   foaf:familyName "Wachkoo" ;
   foaf:givenName "Aijaz␣Ahmad" .
```

LISTING 3.3: .

```
:2b836ad5-db56-4093-9752-33c9f7892de6   rdf:type   fabio:JournalArticle ;
  rdfs:label   "Changes␣to␣publication␣requirements␣made␣at␣the␣XVIII␣Internation\
al␣Botanical␣Congress␣in␣Melbourne␣-␣what␣does␣e-publication␣mean␣for␣you?" ;
  dc:title   "Changes␣to␣publication␣requirements␣made␣at␣the␣XVIII␣International\
␣Botanical␣Congress␣in␣Melbourne␣-␣what␣does␣e-publication␣mean␣for␣you?" ;
 prism:doi   "10.3897/mycokeys.1.1961" ;
 dc:publisher   "Pensoft␣Publishers" ;
 prism:publicationDate   "2011-9-14"^^xsd:date ;
 dcterms:publisher   openbiodiv:0df76aab-1fcf-4118-8e50-198e830a7bed .
 openbiodiv:151a37ba-a337-4855-8e01-200f5ec0251b   rdf:type   deo:Introduction ;
        po:isContainedBy   openbiodiv:2b836ad5-db56-4093-9752-33c9f7892de6 .
}
```

Divide and conquer

After we have successfully converted the current XML node to RDF, a recursive call to Extractor is made for all nodes that are hierarchically dependent on the current node. For example, the article node contains all the other other nodes such as sections, figures, etc.

Transformation specification

In order for the Extractor to work, therefore, we need to specify an XML schema. The specification includes what XML nodes we are looking for and their location. It then recursively specifies for each node, what sub-nodes we are looking for and their XPATH location relative to their parent node. Finally, for every node we need to give the atom locations and write a constructor. The transformation specification is done with R6 framework in R. We have specified two schemata that share the same constructors—TaxPub[4] and TaxonX[5].

3.5.4 Submission to graph database and post-processing

In the previous section we described how we transform XML documents in TaxPub and TaxonX to RDF statements according to OpenBiodiv-O. In addition, we transform the GBIF backbone taxonomy to RDF according to OpenBiodiv-O with the help of TSV4RDF, a proprietary Pensoft tool. The generated RDF statements are submitted to a repository in a GraphDB instance residing on `http://graph.openbiodiv.net/`. The repository has been initialized with OpenBiodiv-O and the ontologies on which it depends[6]. Finally, after the data has been submitted, update scripts are run to generate further statements from our ontology that have not been encoded in OWL for the updating of scientific name relations.

Update rule for replacement name

We state that a scientific name A replaces a scientific name B, if there exists a taxonomic name usage of A with taxonomic status `:ReplacementName` and B is mentioned by a taxonomic name usage in the nomenclatural citations of the treatment, where the discussed taxonomic name usage of A is in the nomenclature section (Listing 3.4).

LISTING 3.4: Update rule for replacement name.

```
PREFIX dwciri: <http://rs.tdwg.org/dwc/iri/>
PREFIX openbiodiv: <http://openbiodiv.net/>
PREFIX po: <http://www.essepuntato.it/2008/12/pattern#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX pkm: <http://proton.semanticweb.org/protonkm#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

INSERT
{
    GRAPH <http://openbiodiv.net/Updates>
    {
    ?name2 openbiodiv:replacementName ?name .
   }
}

WHERE {
        ?tnu1 dwciri:taxonomicStatus openbiodiv:ReplacementName ;
       pkm:mentions ?name.
     ?name dwciri:taxonRank ?rank;
           rdfs:label ?vname .

    ?s po:contains ?tnu .
    ?s po:contains ?citations.
    ?citations rdf:type openbiodiv:NomenclatureCitationsList;
               po:contains ?tnu2 .
    ?tnu2 rdf:type openbiodiv:TaxonomicNameUsage ;
          pkm:mentions ?name2.
    ?name2 rdfs:label ?vname2;
           dwciri:taxonRank ?rank.
}
```

Update rule for related name

The related names update-rule is similar to the replacement name: two scientific names A and B are considered related if they both mentioned in the nomenclature section of a treatment (Listing 3.5).

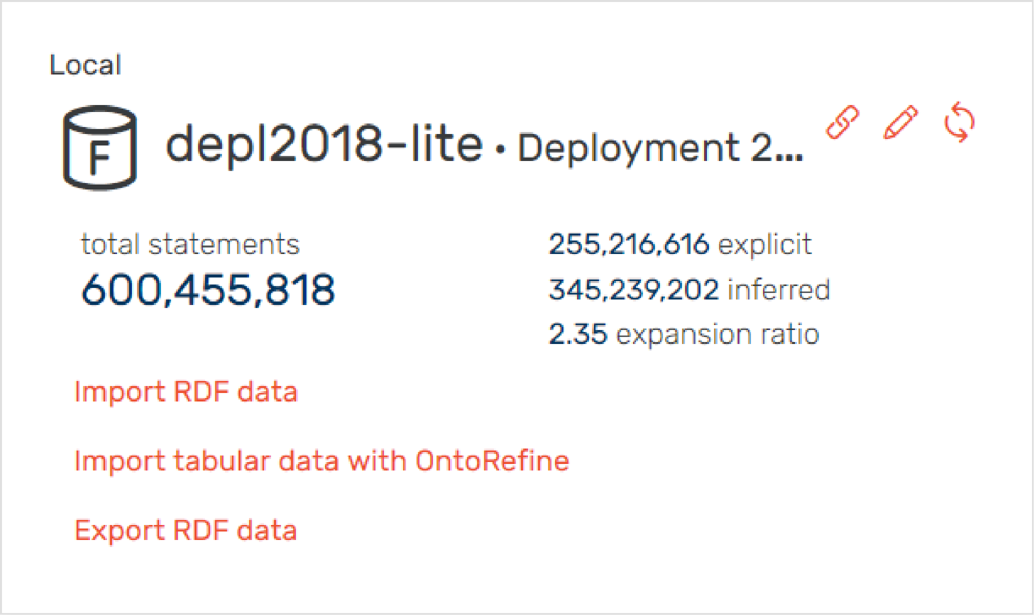
LISTING 3.5: Update rule for related name.

```
PREFIX dwciri: <http://rs.tdwg.org/dwc/iri/>
PREFIX : <http://openbiodiv.net/>
PREFIX po: <http://www.essepuntato.it/2008/12/pattern#>
```

[4] https://github.com/pensoft/ropenbio/blob/redesign/R/taxpub.R

[5] https://github.com/pensoft/ropenbio/blob/redesign/R/taxonx.R

[6] https://github.com/vsenderov/openbiodiv-o/tree/master/imports



Local

F depl2018-lite • Deployment 2...

total statements
600,455,818

255,216,616 explicit
345,239,202 inferred
2.35 expansion ratio

Import RDF data

Import tabular data with OntoRefine

Export RDF data

FIGURE 3.6: Statements report from the GraphDB workbench.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX pkm: <http://proton.semanticweb.org/protonkm#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

INSERT
{
    GRAPH <http://openbiodiv.net/Updates>
    {
    ?name2 :relatedName ?name .
   }
}

WHERE {
  ?nom_sec rdf:type :NomenclatureSection ;
    :contains ?tnu1 .

  ?tnu1 rdf:type :TaxonomicNameUsage ;
    pkm:mentions ?name.

  ?nom_sec :contains ?tnu2 .

  ?tnu2 rdf:type :TaxonomicNameUsage ;
    pkm:mentions ?name2.

  FILTER(?name != ?name2)
}
```

3.6 Performance degradation analysis

The current iteration of the database holds over 600 million triples (Fig. 3.6). The expansion ratio under the RDFS-Plus (Optimized) ruleset is 2.35, i.e. for each asserted statements we materialize on average 2.35 implicit statements. Under the OWL2-RL ruleset (which contains a full implementation of OWL logic rules), the expansion ratio is about 3.7; however, we encountered significant performance issues using it (Fig. 3.7). Even with the lighter ruleset (RDFS-Plus Optimized), we still see performance degradation with increasing database size. Importing the GBIF backbone taxonomy from file takes about two days under the easier scenario. The subsequent importing of the Pensoft archives takes about two weeks as it is a slower operation requiring not only the time for submission but the time for converting the XML's to RDF.

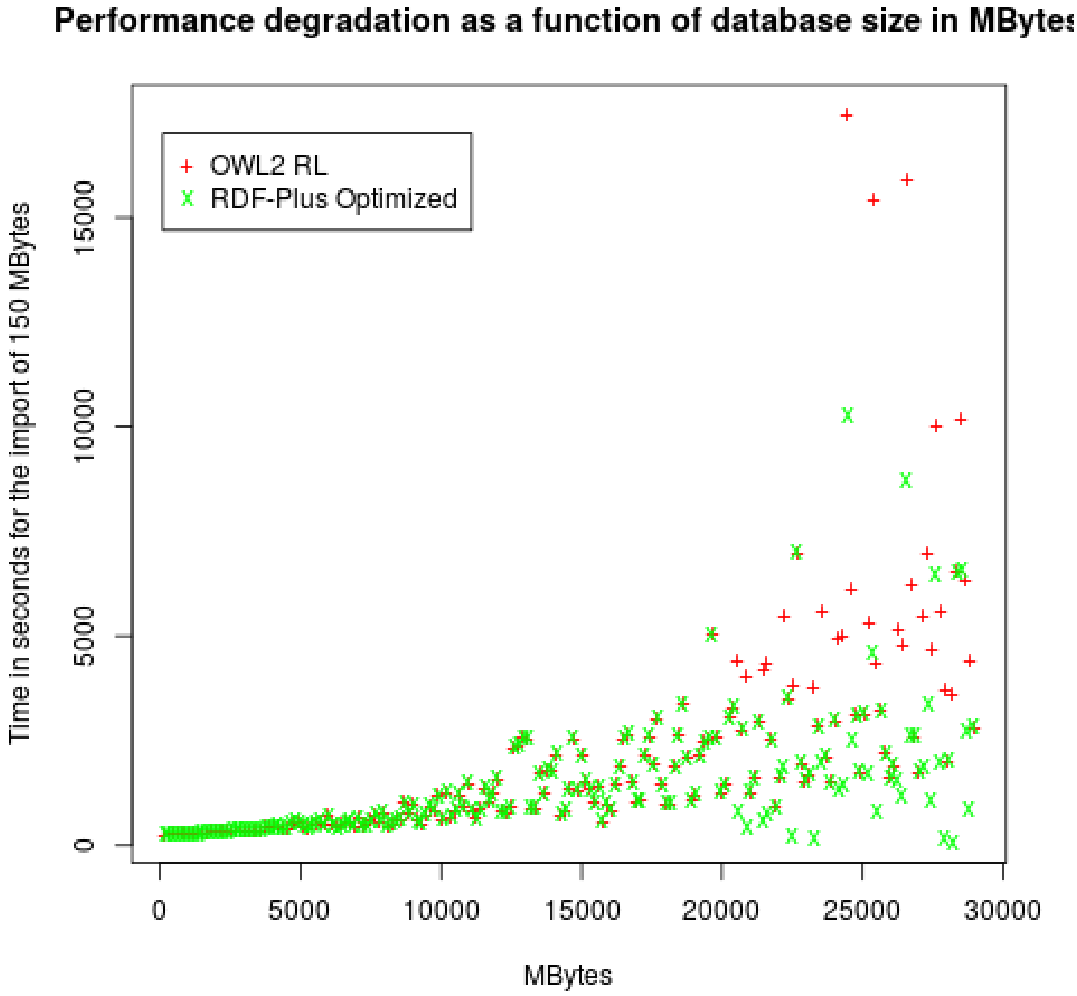


Figure 3.7: The graph visualizes the time in seconds needed to import a 150 MB big Turtle data file as a function of the database size. The database size is measured by the adding up the size of the data files that have already been imported.

Chapter 4

Summary of Chapter 4: An R Library for Working with RDF

RDF4R (`rdf4r`) is an R package for working with Resource Description Framework (RDF Working Group, 2014) data. It was developed as part of the OpenBiodiv project but is completely free of any OpenBiodiv-specific code and can be used for generic purposes requiring tools to work with RDF data in the R programming environment (R Core Team, 2016).

4.1 Installation

In this section we describe how to install the RDF4R package. Installation is straighforward and consists of two steps: (1) resolve dependencies and (2) build the package from source using `devtools::install_github`.

4.2 Specification

In this section we present the specifications of RDF4R by detailing the features of the package. Each feature has a dedicated subsection.

4.2.1 Connection to a triple-store

It is possible to establish both basic connections (requiring no password or requiring basic HTTP user-pass authentication) or connection secured with an API access token.

4.2.2 Work with repositories on a triple-store

Once a connection to a triple-store has been established, it is possible to inspect the talk protocol version, view the list of repositories on the database, execute SPARQL Read (SELECT keyword and related) and SPARQL Update (INSERT and related) queries on the database, as well as submit serialized RDF data directly to the database.

4.2.3 Function factories to convert SPARQL queries to R functions

An important feature of RDF4R are its facilities for converting SPARQL queries and the like to R functions.

4.2.4 Work with literals and identifiers

The building blocks of RDF are literals (e.g. strings, numbers, dates, etc.) and resource identifiers. RDF4R provides classes for literals and resource identifiers that are tightly integrated with the other facilities of the package.

4.2.5 Prefix management

Prefixes are managed automatically during serialization by being extracted from the resource identifiers.

4.2.6 Creation and serialization of RDF

The serialization function supports Turtle (and its variant Trig, Bizer and Cyganiak, 2014) and adding new triples.

LISTING 4.1: Using brackets to express RDF blank nodes in Turtle/TriG.

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

# :someone knows someone else, who has the name "Bob".
:someone foaf:knows [ foaf:name "Bob" ] .
```

4.2.7 A basic vocabulary of semantic elements

RDF4R has some basic resource identifiers for widely used classes and predicates predefined (e.g. for `rdf:type`, `rdfs:label`, etc.).

4.3 Usage

Here, we explain how to use the package RDF4R by means of examples. In order to fully utilize the package capabilities, one needs to have access to an RDF graph database. We have made available a public endpoint (see next paragraph) to allow the users of the package to experiment. Since write access is enabled, please be considerate and don't issue catastrophic commands.

4.4 Discussion

4.4.1 Related Packages

The closest match to RDF4R is the `rdflib` (Boettiger, 2018). The development of the two packages was simultaneous and independent until `rdflib`'s first official release on Dec 10, 2017. This explains why two closely related R packages for working with RDF exist. After the release of `rdflib` work was started to make both packages compatible with each other. In our opinion, the packages have different design philosophies and are thus complementary.

`rdflib` is a high-level wrapper to `redland` (Jones et al., 2016), which is a low-level wrapper to the C `librdf` (Beckett, 2014), a powerful C library that provides support for RDF. `librdf` provides an in-memory storage model for RDF beyond what is available in RDF4R and also persistent storage working with a number of databases. It enables the user to query RDF objects with SPARQL. Thus, `librdf` can be considered a complete graph database implementation in C.

In our opinion, `redland` is more complex than needed for the purposes of OpenBiodiv. By the onset of the OpenBiodiv project it was available[1]; however, we decided not to use it as a decision was made to rely on GraphDB for our storage and querying. Note that RDF4R's main purpose is to provide a convenient R interface for users of GraphDB and similar RDF4J compatible graph databases.

[1] But not `rdflib`!

A feature that differentiates `rdflib` from RDF4R is the design philosophy. RDF4R was designed primarily with the Turtle and TriG serializations in mind. This means that RDF4R can work with named graphs, whereas their usage is discouraged or perhaps impossible with `rdflib`[2], even though `rdflib`'s default format is N-Quads.

Another differentiating feature between RDF4R and `rdflib` is that RDF4R provides facilities for converting SPARQL and related statements to native R functions!

In a future release of RDF4R (2.0) we would like to replace or extend its in-memory model with `rdflib`'s. This is why we would like to make the packages fully compatible and have contributed several patches to `rdflib`[3]). Thus, it will be possible for the user of RDF4R to retain its syntax and high-level features— constructor factories, functors, etc., and the ability to use named graphs—but benefit from performance increases, stability, and scalability with the `redland/rdflib/librdf` backend.

This will enable the users of the R programming environment to use whichever syntax they prefer and benefit from an efficient storage engine.

4.4.2 Elements of Functional Programming (FP)

In this subsection we discuss how patterns from functional programming were used to create RDF4R.

4.4.3 Elements of Object-Oriented Programming (OOP)

In this subsection we discuss how patterns from object-oriented programming were used to create RDF4R.

[2] The issue was discussed on the `librdf` GitHub page, `https://github.com/ropensci/rdflib/issues/23`.

[3] Please, consult the commit history under `https://github.com/ropensci/rdflib`.

Chapter 5

Summary of Chapter 5: Workflows for Biodiversity Data

In this chapter we discuss two automated workflows for exchange of biodiversity data developed as part of OpenBiodiv: (1) automatic import of specimen records into manuscripts, and (2) automatic generation of data paper manuscripts from Ecological Metadata Language (EML) metadata. The workflows were presented at a webinar for the orgnization iDigBio[1] and published as a paper (Senderov et al., 2016).

The slides from the presentation as well as a PDF of the paper are available from the webinar GitHub page under `https://github.com/vsenderov/idigbio-webinar`.

5.1 Introduction

Information on occurrences of species and information on the specimens that are evidence for these occurrences (specimen records) is stored in different biodiversity databases. These databases expose the information via public REST API's. I focused on the Global Biodiversity Information Facility (GBIF), Barcode of Life Data Systems (BOLD), iDigBio, and PlutoF, and utilized their API's to import occurrence or specimen records directly into a manuscript edited in the ARPHA Writing Tool (AWT).

Furthermore, major ecological and biological databases around the world provide information about their datasets in the form of EML. A workflow was developed for creating data paper manuscripts in AWT from EML files. Such files could be downloaded, for example, from GBIF, DataONE, or the Long-Term Ecological Research Network (LTER Network).

The development of these workflows focuses on two areas: optimizing the workflow of specimen data and optimizing the workflow of dataset metadata. These efforts resulted in the functionality that it is now possible, via a record identifier, to directly import specimen record information from the Global Biodiversity Information Facility (GBIF), Barcode of Life Data Systems (BOLD), iDigBio, or PlutoF into manuscripts in the ARPHA Writing Tool (AWT). No manual copying or retyping is required.

[1] Integrated Digitized Biocollections (iDigBio) is a US-based aggregator of biocollections data. They hold regular webinars and workshops aimed at improving biodiversity informatics knowledge, which are attended by collection managers, scientists, and IT personnel. Thus, doing a presentation for iDigBio was an excellent way of making the research and tools-development efforts of OpenBiodiv widely known and getting feedback from the community.

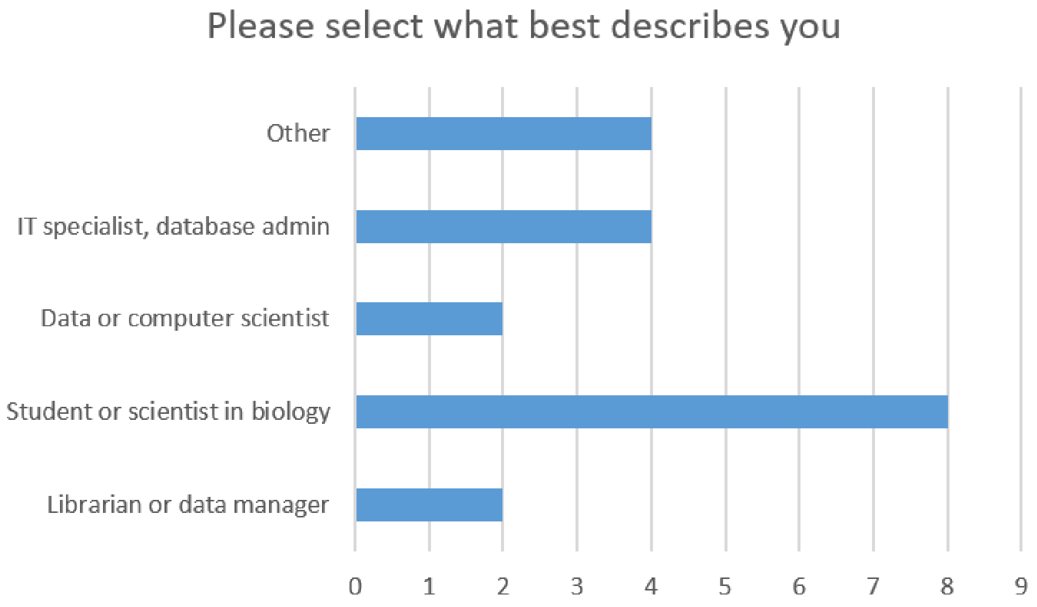


FIGURE 5.1: Poll results about composition of audience during live participation..

5.2 Presentation

A video recording of the presentation is available[2]. More information can be found in the webinar information page[3]. The slides of the presentation are attached as supplementary files and are deposited in Slideshare[4].

During the presentation we conducted a poll about the occupation of the attendees, the results of which are summarized in Fig. 5.1. Of the participants who voted, about a half were scientists, mostly biologists, while the remainder were distributed across IT specialists and librarians, with 20% "Other." The other categories might have been administrators, decision-makers, non-biology scientists, collections personnel, educators, etc.

At the end of the presentation, very interesting questions were raised and discussed. For details, see the “Results and discussion” section of this paper.

5.3 Methods

Both workflows discussed rely on three key standards: RESTful API’s for the web (Kurtz, 2013), Darwin Core (Wieczorek et al., 2012), and EML (Fegraus et al., 2005).

5.3.1 Development of workflow 1: Automated specimen record import

In this subsection we discuss the development of Workflow 1: Automated specimen record import.

[2] http://idigbio.adobeconnect.com/p7sg0aym3e3/
[3] http://www.idigbio.org/content/online-direct-import-specimen-records-idigbio-infrastructure-taxonomic-mar
[4] http://www.slideshare.net/ViktorSenderov/online-direct-import-of-specimen-records-from-idigbio-infrastruc

5.3.2 Development of workflow 2: Automated data paper generation

In this subsection we discuss the development of Workflow 1: Automated specimen record import.

5.4 Results and Discussion

5.4.1 Workflow 1: Automated specimen record import into manuscripts developed in the ARPHA Writing Tool

It is now possible to directly import a specimen record as a material citation in an ARPHA Taxonomic Paper from GBIF, BOLD, iDigBio, and PlutoF (Slide 5, as well as Fig. 5.2). The workflow from the user's perspective has been thoroughly described in a blog post; concise stepwise instructions are available via ARPHA's Tips and tricks guidelines. In a nutshell, the process works as follows:

1. At one of the supported data portals (BOLD, GBIF, iDigBio, PlutoF), the author locates the specimen record he/she wants to import into the Materials section of a Taxon treatment (available in the Taxonomic Paper manuscript template).

2. Depending on the portal, the user finds either the occurrence identfier of the specimen, or a database record identifier of the specimen record, and copies that into the respective upload field of the ARPHA system (Fig. 5.3).

3. After the user clicks on "Add," a progress bar is displayed, while the specimens are being uploaded as material citations.

4. The new material citations are rendered in both human- and machine-readable DwC format in the Materials section of the respective Taxon treatment and can be further edited in AWT, or downloaded from there as a CSV file.

Discussion

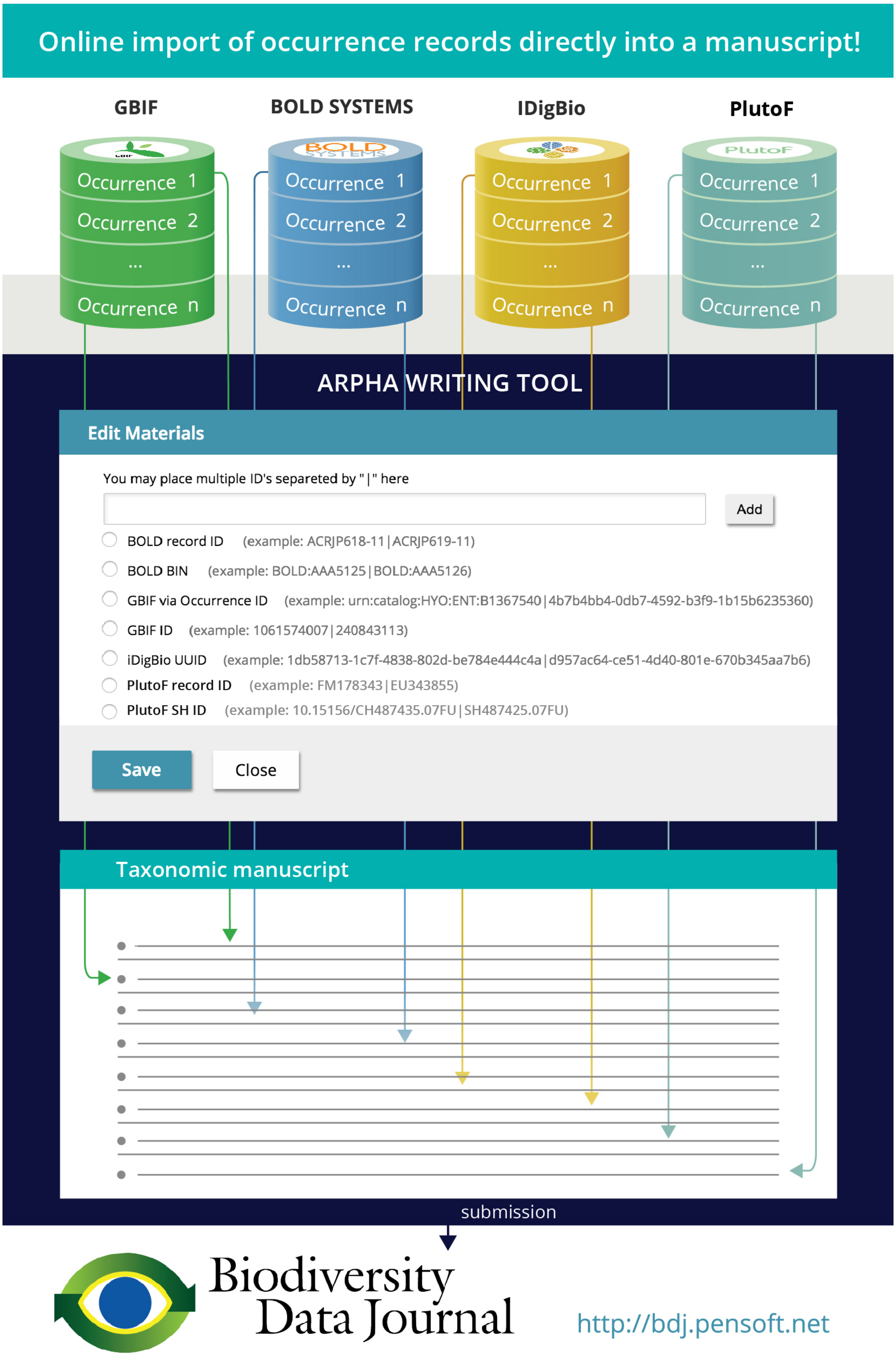
We discuss the availability, or more correctly the lack of persistent unique identifiers (PID's) in the biodiversity informatics space. I furthermore discuss the challenges of importing from our different sources: GBIF, PlutoF, iDigBio, and BOLD. I emphasize how our workflow can be serve as a curation filter for increasing the quality of specimen data via the scientific peer review process.

5.4.2 Workflow 2: Automated data paper manuscript generation from EML metadata in the ARPHA Writing Tool

We have created a workflow that allows authors to automatically create data paper manuscripts from the metadata stored in EML (Fig. 5.4, Fig. 5.5, Fig. 5.6).

Discussion

I discuss the history of data papers and how our implementation greatly improves the availability of data papers to science practicioners. The two workflows presented generated a lively discussion at the end of the presentation, which is summarized in the Chapter.



You may place multiple ID's separeted by "|" here

○ BOLD record ID (example: ACRJP618-11|ACRJP619-11)

○ BOLD BIN (example: BOLD:AAA5125|BOLD:AAA5126)

○ GBIF via Occurrence ID (example: urn:catalog:HYO:ENT:B1367540|4b7b4bb4-0db7-4592-b3f9-1b15b6235360)

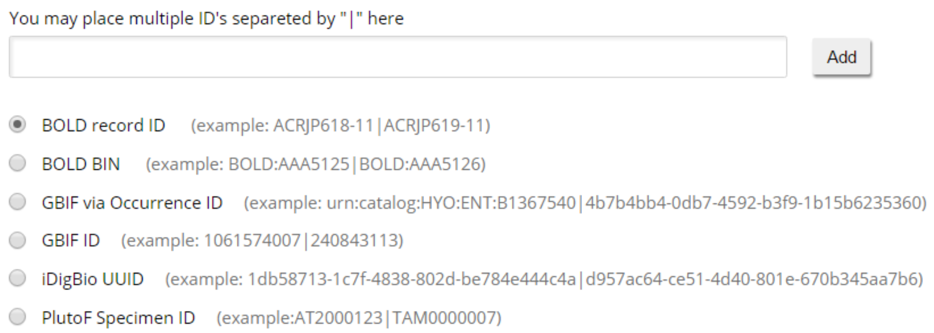
○ GBIF ID (example: 1061574007|240843113)

○ iDigBio UUID (example: 1db58713-1c7f-4838-802d-be784e444c4a|d957ac64-ce51-4d40-801e-670b345aa7b6)

○ PlutoF record ID (example: FM178343|EU343855)

○ PlutoF SH ID (example: 10.15156/CH487435.07FU|SH487425.07FU)

FIGURE 5.2: This fictionalized workflow presents the flow of information content of biodiversity specimens or biodiversity occurrences from the data portals GBIF, BOLD Systems, iDigBio, and PlutoF, through user-interface elements in AWT to textualized content in a Taxonomic Paper manuscript template intended for publication in the Biodiversity Data Journal.



You may place multiple ID's separeted by "|" here

Add

● BOLD record ID (example: ACRJP618-11|ACRJP619-11)

○ BOLD BIN (example: BOLD:AAA5125|BOLD:AAA5126)

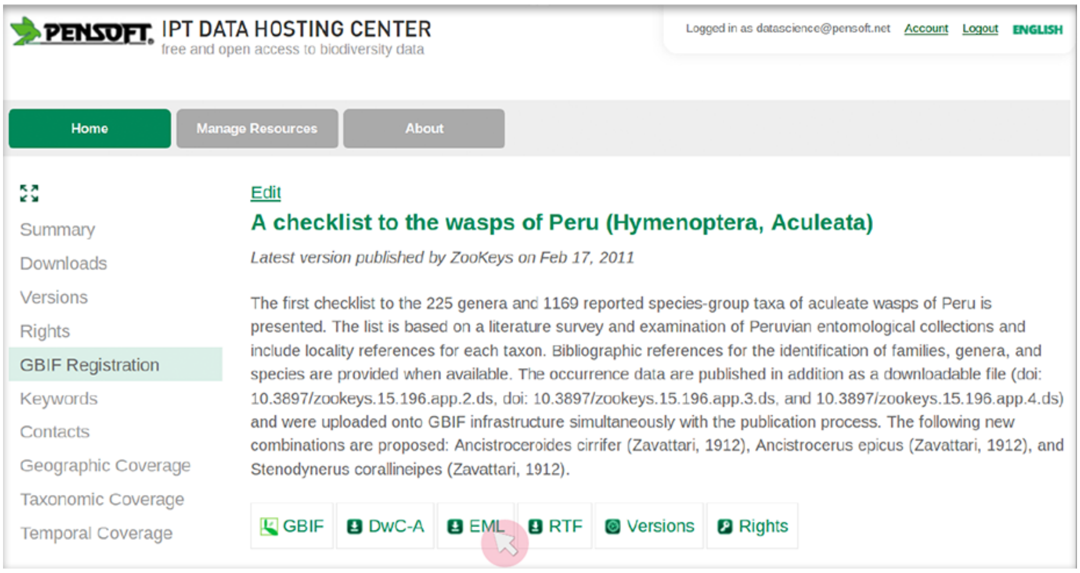
○ GBIF via Occurrence ID (example: urn:catalog:HYO:ENT:B1367540|4b7b4bb4-0db7-4592-b3f9-1b15b6235360)

○ GBIF ID (example: 1061574007|240843113)

○ iDigBio UUID (example: 1db58713-1c7f-4838-802d-be784e444c4a|d957ac64-ce51-4d40-801e-670b345aa7b6)

○ PlutoF Specimen ID (example:AT2000123|TAM0000007)

FIGURE 5.3: User interface of the ARPHA Writing Tool controlling the import of specimen records from external databases.



PENSOFT. IPT DATA HOSTING CENTER
free and open access to biodiversity data

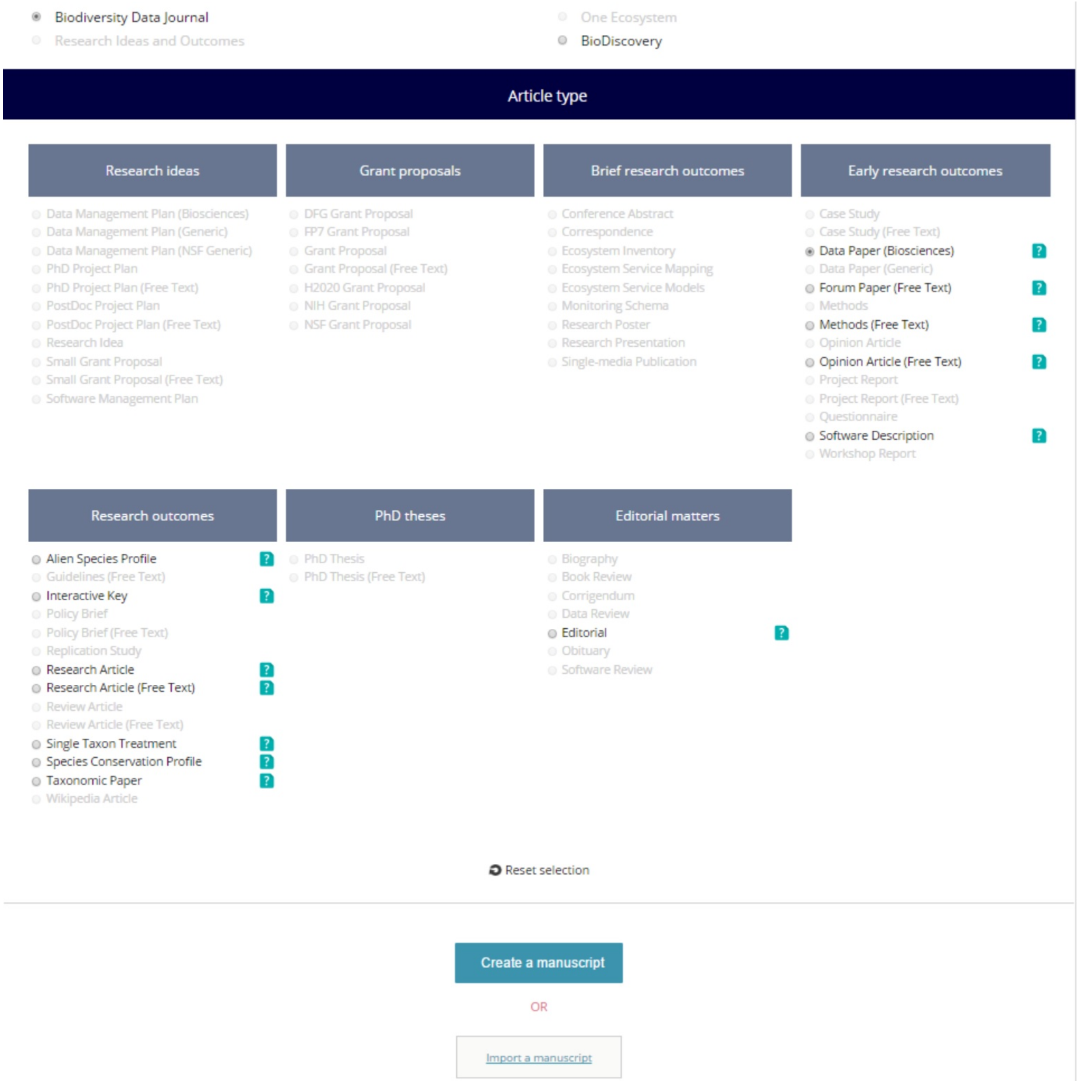
Logged in as datascience@pensoft.net Account Logout ENGLISH

A checklist to the wasps of Peru (Hymenoptera, Aculeata)

Latest version published by ZooKeys on Feb 17, 2011

The first checklist to the 225 genera and 1169 reported species-group taxa of aculeate wasps of Peru is presented. The list is based on a literature survey and examination of Peruvian entomological collections and include locality references for each taxon. Bibliographic references for the identification of families, genera, and species are provided when available. The occurrence data are published in addition as a downloadable file (doi: 10.3897/zookeys.15.196.app.2.ds, doi: 10.3897/zookeys.15.196.app.3.ds, and 10.3897/zookeys.15.196.app.4.ds) and were uploaded onto GBIF infrastructure simultaneously with the publication process. The following new combinations are proposed: Ancistroceroides cirrifer (Zavattari, 1912), Ancistrocerus epicus (Zavattari, 1912), and Stenodynerus corallineipes (Zavattari, 1912).

FIGURE 5.4: Download of an EML from the GBIF Integarted Publishuing Toolkit (IPT).



● Biodiversity Data Journal
○ Research Ideas and Outcomes
○ One Ecosystem
○ BioDiscovery

Article type

Research ideas

- ○ Data Management Plan (Biosciences)
- ○ Data Management Plan (Generic)
- ○ Data Management Plan (NSF Generic)
- ○ PhD Project Plan
- ○ PhD Project Plan (Free Text)
- ○ PostDoc Project Plan
- ○ PostDoc Project Plan (Free Text)
- ○ Research Idea
- ○ Small Grant Proposal
- ○ Small Grant Proposal (Free Text)
- ○ Software Management Plan

Grant proposals

- ○ DFG Grant Proposal
- ○ FP7 Grant Proposal
- ○ Grant Proposal
- ○ Grant Proposal (Free Text)
- ○ H2020 Grant Proposal
- ○ NIH Grant Proposal
- ○ NSF Grant Proposal

Brief research outcomes

- ○ Conference Abstract
- ○ Correspondence
- ○ Ecosystem Inventory
- ○ Ecosystem Service Mapping
- ○ Ecosystem Service Models
- ○ Monitoring Schema
- ○ Research Poster
- ○ Research Presentation
- ○ Single-media Publication

Early research outcomes

- ○ Case Study
- ○ Case Study (Free Text)
- ● Data Paper (Biosciences) ?
- ○ Data Paper (Generic)
- ○ Forum Paper (Free Text) ?
- ○ Methods
- ○ Methods (Free Text) ?
- ○ Opinion Article
- ○ Opinion Article (Free Text) ?
- ○ Project Report
- ○ Project Report (Free Text)
- ○ Questionnaire
- ○ Software Description ?
- ○ Workshop Report

Research outcomes

- ○ Alien Species Profile ?
- ○ Guidelines (Free Text)
- ○ Interactive Key ?
- ○ Policy Brief
- ○ Policy Brief (Free Text)
- ○ Replication Study
- ○ Research Article ?
- ○ Research Article (Free Text) ?
- ○ Review Article
- ○ Review Article (Free Text)
- ○ Single Taxon Treatment ?
- ○ Species Conservation Profile ?
- ○ Taxonomic Paper ?
- ○ Wikipedia Article

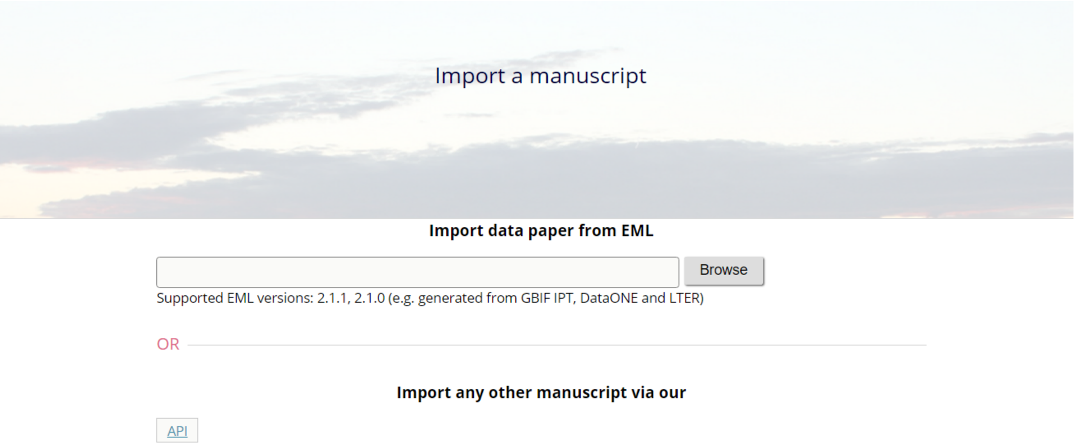
PhD theses

- ○ PhD Thesis
- ○ PhD Thesis (Free Text)

Editorial matters

- ○ Biography
- ○ Book Review
- ○ Corrigendum
- ○ Data Review
- ○ Editorial ?
- ○ Obituary
- ○ Software Review

FIGURE 5.5: Selection of the journal and "Data Paper (Biosciences)" template in the ARPHA Writing Tool.



Import a manuscript

Import data paper from EML

Supported EML versions: 2.1.1, 2.1.0 (e.g. generated from GBIF IPT, DataONE and LTER)

OR

Import any other manuscript via our

FIGURE 5.6: The user interface field for uploading EML files into ARPHA.

Chapter 6

Summary of Chapter 6: Web portal

Under openbiodiv.net one can reach the main portal giving access to OpenBiodiv resources. This portal was developed by Pensoft to support OpenBiodiv. OpenBiodiv.net presents two visual elements to the user: the search bar and list of application icons in the bottom. Furthermore, under graph.openbiodiv.net (also accessible from the icon SPARQL endpoint) one can reach the OpenBiodiv workbench, a feature of GraphDB that gives web access to the SPARQL endpoint.

These User Interface (UI) features are designed to facilitate the three user types of the system that we envisage:

1. Basic level: uses search bar.
2. Specialist level: uses apps.
3. Power user: uses the work-bench of the system or R.

6.1 Functionality of the system

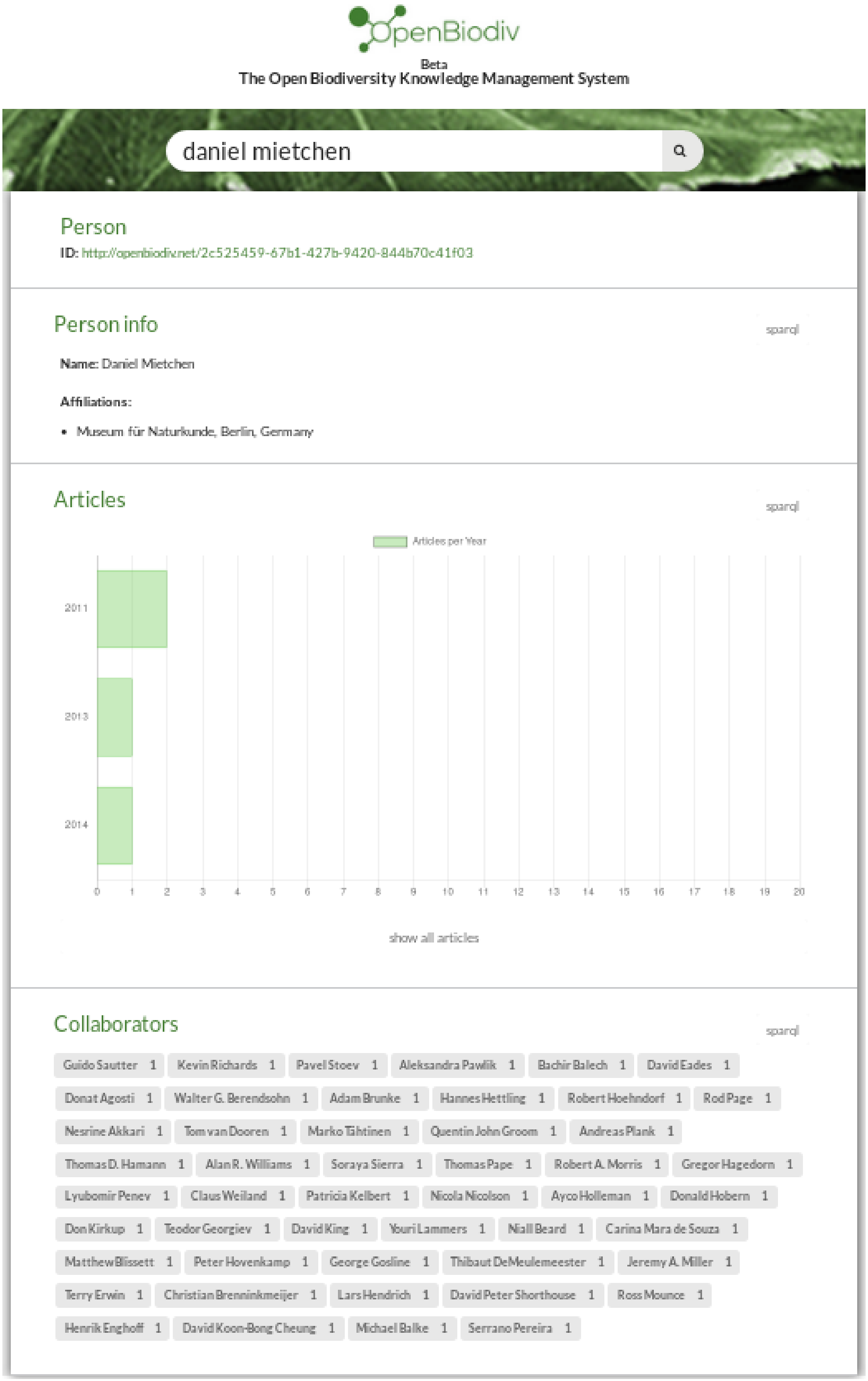
In this section we discuss how every user-type can use the system.

6.1.1 Basic usage

The basic level of interaction is for users who want a quick look into the system's database; they can be beginners without knowledge of the Semantic Web or of taxonomy, or advanced users with little time or a very basic query. An example of such a user will simply look for an entity (e.g. taxonomic name, person) and would like to retrieve some information about it.

6.1.2 Specialist level

A specialist is someone who has a question of particular taxonomic importance that cannot be answered by a simple name-based look-up. For example, a collection manager at a museum may want to periodically check for articles that make use of their collection in order to justify additional funding to prevent natural disasters. Or a taxonomist interested in a particular region or group may want to stay up to date with published literature fitting those criteria—let's say weevils (Curculionidae) of Arizona, U.S.A.



OpenBiodiv

Beta

The Open Biodiversity Knowledge Management System

daniel mietchen

Person

ID: http://openbiodiv.net/2c525459-67b1-427b-9420-844b70c41f03

Person info

sparql

Name: Daniel Mietchen

Affiliations:

- Museum für Naturkunde, Berlin, Germany

Articles

sparql

show all articles

Collaborators

sparql

Guido Sautter	1
Kevin Richards	1
Pavel Stoev	1
Aleksandra Pawlik	1
Bachir Balech	1
David Eades	1
Donat Agosti	1
Walter G. Berendsohn	1
Adam Brunke	1
Hannes Hettling	1
Robert Hoehndorf	1
Rod Page	1
Nesrine Akkari	1
Tom van Dooren	1
Marko Tähtinen	1
Quentin John Groom	1
Andreas Plank	1
Thomas D. Hamann	1
Alan R. Williams	1
Soraya Sierra	1
Thomas Pape	1
Robert A. Morris	1
Gregor Hagedorn	1
Lyubomir Penev	1
Claus Weiland	1
Patricia Kelbert	1
Nicola Nicolson	1
Ayco Holleman	1
Donald Hobern	1
Don Kirkup	1
Teodor Georgiev	1
David King	1
Youri Lammers	1
Niall Beard	1
Carina Mara de Souza	1
Matthew Blissett	1
Peter Hovenkamp	1
George Gosline	1
Thibaut DeMeulemeester	1
Jeremy A. Miller	1
Terry Erwin	1
Christian Brenninkmeijer	1
Lars Hendrich	1
David Peter Shorthouse	1
Ross Mounce	1
Henrik Enghoff	1
David Koon-Bong Cheung	1
Michael Balke	1
Serrano Pereira	1

FIGURE 6.1: Illustration of basic usage of OpenBiodiv to look information about a person.

6.1.3 Power user

The power user is someone with knowledge of the Semantic Web and its technologies (SPARQL, ontologies, etc.). The power user goes to the workbench and executes their queries there, or uses the functionality of the RDF4R package described in Chapter 4 to execute SPARQL directly on the OpenBiodiv endpoint directly from the R environemnt.

6.2 Implementation

The UI-components of the web portal are developed in the ReactJS JavaScript framework written by Facebook. Server-side processing is done in PHP. This part of OpenBiodiv is not open source and cannot be discussed in detail in the present dissertation effort.

Conclusion

Results

We believe that the presented scientific work fulfills the stated objective and tasks.

Result 1. The central result of the thesis is the creation of a domain conceptualization of biodiversity publishing and a formal ontology OpenBiodiv-O enabling the linking of biodiversity knowledge on the basis of scholarly publications. This result has been described in Chapter 2 and in Senderov et al., 2018 and fulfills Objective 1. The source code of the ontology is available under github.com/pensoft/openbiodiv-o.

Result 2. The second result of the thesis is the creation of the software architecture of the OpenBiodiv system outlined in Chapter 1 and Senderov and Penev, 2016. This result fulfills Objective 2.

Result 3. The third result of the thesis has been the creation of a Linked Open Dataset, OpenBiodiv-LOD, consisting of a transformation to RDF-triples and integration in a single store of information from three major repositories of biodiversity data: the XML sources of biological journals published by Pensoft Publishers, the XML sources of treatments freed by Plazi, and a CSV dump of GBIF's taxonomic backbone. OpenBiodiv-LOD is available under `graph.openbiodiv.net` and has been described in Chapter 3. This result fulfills Objective 3.

Result 4. In order to create the Linked Open Data, a software package for the R programming environment, RDF4R, was developed. RDF4R enables the manipulation of RDF data within R and facilities the transformation of scientific publications from a semi-structured XML format to structured semantic RDF. This result has been discussed in Chapter 4 and fulfills Objective 4. The package is available online as free software under `github.com/pensoft/rdf4r`. Furthermore, additional source code (unoptimized) describing XML schemata of Pensoft and Plazi and working in tandem with RDF4R to convert XML to RDF can be found under `github.com/pensoft/ropenbio`.

Result 5. The mechanisms to convert semi-structured XML into RDF-triples are complemented by workflows enabling the enrichment of the XML sources of Pensoft journals by data automatically imported from the major international biodiversity data repositories: BOLD, GBIF, iDigBio, as well as PlutoF. Furthermore, it is now possible, thanks to this dissertation effort to automatically create manuscripts from metadata encoded in the Ecological Metadata Language (EML). The discussion of these automated workflows—automatic data paper generation and automatic occurrence record import—is carried out in Chapter 5. It fulfills Objective 5.

Result 6. To complement the creation of OpenBiodiv-LOD, we have developed a website running on top of the knowledge graph openbiodiv.net, containing a semantic search engine and apps. The website is discussed in Chapter 6 and fulfills Objective 6.

Discussion, conclusion, and outlook

OpenBiodiv-O serves as the basis of the Linked Open Data OpenBiodiv-LOD. By developing an ontology focusing on biological taxonomy, we provided an ontology that fills in the gaps between ontologies for biodiversity resources such as Darwin-SW and semantic publishing ontologies such as the ontologies comprising the SPAR Ontologies. Moreover, we take the view that it is advantageous to model the taxonomic process itself rather than any particular state of knowledge. At this stage, the coverage of the ontology of the different types of resources is sufficient to be the basis for creating the LOD. In this sense, it is completed. On the other hand, adding classes and properties for new types of biodiversity data is possible and desirable.

The LOD, similar to the ontology, are already a solid resource for biologists, as they include information from most articles published by Pensoft and Plazi and count over 600 million triplets. Like the ontology, they should be expanded.

Since the RDF4R package was successfully used to create an LOD, it can be considered complete. Like any software package, however, it should be maintained and developed.

The website is still in beta. The functionality that works great is the semantic search engine. For some basic data types there are templates for visualization. However, the site can not be considered complete and most users use the SPARQL search language.

An important conclusion that can be drawn from the work is that it is possible to use a semantic graph for the integration of a large volume of data on biodiversity. We were unexpectedly given the opportunity to illustrate the power of the knowledge graph by analyzing the damage from the tragic fire at the Museu Nacional in Rio de Janeiro. In addition, we have illustrated that it is possible to write relatively simple logical conclusions to check the validity of a taxonomic name.

Due to the large amount of data, we found that although the use of a semantic graph was possible, some of the initially chosen technologies proved to be inapplicable or difficult to apply. We have observed (see Chapter 3) that the practical application of the full logical OWL model is difficult due to performance problems. Instead in the end, we utilized RDFS that is less powerful but faster. Another observation of ours is that although the R programming environment has given us some advantages in rapidly creating the prototype of the system, by increasing the complexity of the program code needed in the real-life system to cover all private cases, a language with dynamic types such as R creates headaches in debugging. At the same time, we were impressed by the powerful functional programming toolkit R provided.

A big difficulty was the disambiguation of resources such as author names or taxonomic names. In the functional design of the RDF4R package we have put modules that allow us to insert a list of functions/rules for disambiguation when searching for an identifier for a given resource. However, we had only limited success with the rule-based disambiguation and for this reason in the production system it was discontinued at the moment.

Considering these and other “lessons,” the future development of the OpenBiodiv project can be outlined in the following not necessarily comprehensive way:

1. As an immediate goal, to expand the LOD and ontology with new data types and new data sources using the existing framework. Such data are e.g. genomic data, occurrence data, (bio-)geographic data, visual data, descriptive data, etc.
2. Look for even closer integration with other existing biodiversity data repositories than GBIF. For example, BioImages, iNaturalist, BOLD, and so on.
3. As a longer-term task to study the transition from a semantic graph to a technology where the inference engine is separated from the data base layer as WikiData or Neo4j. In addition to increased performance, this will give extra flexibility to the project, such as allowing the use of non-RDF-based inference engines such as Euler.
4. Continue developing system software with an even wider application of functional programming and porting it into a functional language like, for example, Haskell or O'CAML.
5. To investigate the problem of disambiguation and related problems for named entity recognition of interesting resources from biodiversity, as well various image recognition tasks, from the point of view of machine learning.
6. Expanding the website with more templates and new applications.

Key scientific and applied contributions

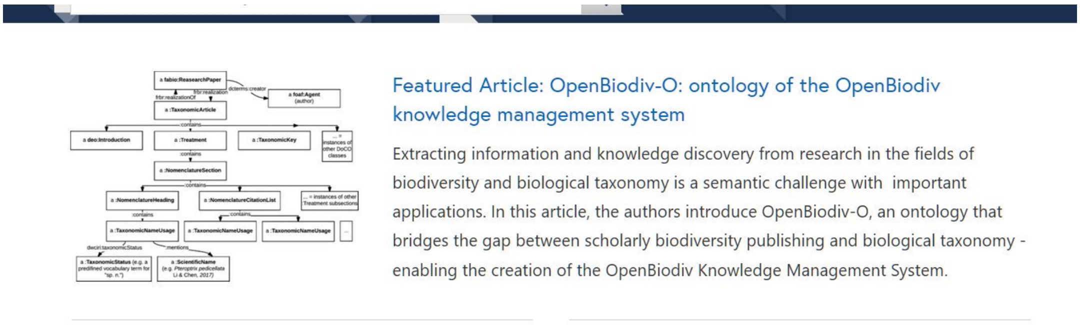
The results discussed in the previous two sections determine the following scientific and applied contributions:

1. Scientific contribution: creating an ontology and a formal model of the field of biodiversity knowledge publication.
2. Applied scientific contribution: analyzing information sources and Creating OpenBiodiv-LOD.
3. Applied scientific Contribution: the implementation of OpenBiodiv software modules.

Our ontology fills the unique niche between bibliographic ontologies such as SPAR and ontologies for biodiversity such as Darwin-SW and as such is undoubtedly of great scientific interest to the biodiversity informatics community. The work has a serious scientific and applied character by providing both a Linked Open Dataset on top of the ontology and software for its users and system developers.

Evaluation of publications

Articles have been published without exception in four international scientific journals: five articles in Research Ideas and Outcomes, one article in ZooKeys (WoS IF 1.079, Q3 SCOPUS, SJR 0.533), one article in Biodiversity Data Journal (WoS SCOPUS, SJR 0.465) and one article in Journal of Biomedical Semantics (WoS IF 1.6, Q3 SCOPUS, SJR 0.952). The total number of citations that have been accumulated for the candidate excluding self-citations (cross-citations) is at least 20. The citing articles are given in the list above. The total number of citations that have been



Featured Article: OpenBiodiv-O: ontology of the OpenBiodiv knowledge management system

Extracting information and knowledge discovery from research in the fields of biodiversity and biological taxonomy is a semantic challenge with important applications. In this article, the authors introduce OpenBiodiv-O, an ontology that bridges the gap between scholarly biodiversity publishing and biological taxonomy - enabling the creation of the OpenBiodiv Knowledge Management System.

Figure 6.2: The OpenBiodiv-O article is featured on the main webpage of the Journal of Biomedical Semantics..

accumulated including cross-citations and citations of work outside of the scope of the dissertation is at least 48 (Google Scholar).

[1] is an early version of the Introduction as well Chapter 1 and contains work towards Objective 2 (Architecture). The text of publications [2, 3, 5, 6, 7] are not a part of the text of the dissertation one-to-one but contain work towards Objective 5 (Workflows). The ideas presented in these publications have to large degree been incorporated in Chapter 5 whose backbone is formed by [4]; thus Objective 5 (Workflows) is achieved. [7] is published in the peer-reviewed journal ZooKeys with impact factor 1.031 (early 2018). [8] is the most important publication under this dissertation and was published in the high-impact Journal of Biomedical Semantics with impact factor 2.413 (early 2018). [8] makes up the content of Chapter 2 and is the main body of work fulfilling Objective 1 (Ontology). It was a featured article on the home-page of JBS (Fig. 6.2). Chapter 3 and Chapter 4 that form Objectives 3, 4, respectively are currently being prepared as manuscripts in international journals. Furthermore, the software library RDF4R described in Chapter 4 is being submitted to the open source repository rOpenSci[1].

[1] "We build software with a community of users and developers, and educate scientists about transparent research practices." https://ropensci.org/

Acknowledgements

This research has been financed through the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No. 642241. My deep gratitude goes to the European Commission for enabling this wonderful opportunity!

I thank Prof. Lyubomir Penev and Prof. Kiril Simov for the valuable supervision. I also thank the staff and developers at Pensoft Publishers for the support in creating the platform and its popularization; in particular Prof. Pavel Stoev, Teodor Georgiev, Georgi Zhelezov, Iliyana Kuzmova, and Iva Kostadinova. Furthermore, I thank Pensoft's graphic designer, Slavena Peneva, for the help with creating the illustrations for this thesis and in presentations. Last but not least, Margarita Grudova and Elisaveta Taseva for providing valuable administrative support during the elaboration of the thesis.

I thank my colleagues from the Bulgarian Academy of Sciences (Institutes for Information and Communication Technologies and for Biodiviversity and Ecosystems Research) for their friendship and advice; in particular Prof. Galya Angelova, Prof. Boyko Georgiev, and Prof. Snejana Grozeva.

I thank my colleagues at the BIG4 training network for the feedback, friendship, and support. In particular Prof. Alexey Solovdnikov, but there are too many more names to mention.

I thank my international collaborators for their ideas, reviews, and collaboration on papers. In particular Prof. Nico Franz (Arizona State University), Dr. Daniel Mietchen (National Institutes of Health), Dr. Éamonn Ó Tuama (formerly at GBIF), and Prof. Bob Morris (emeritus UMASS).

I also thank everyone at Plazi for the co-ownership of the vision of the project; in particular, Dr. Donat Agosti, Terry Catapano, and Dr. Guido Sautter.

Last but not least, I would like to acknowledge Ontotext for building the GraphDB database and providing excellent support.