OneForestKB: A Knowledge Base for Global Forest Data Curation and Exploitation Using Geospatial Services

Felipe Vargas-Rojas [a,*], Vincent Armant [a] and Isabelle Mougenot [a]
[a] *ESPACE-DEV, IRD, Université de Montpellier, Université Antilles, Université de Guyane, Université de la Réunion, Montpellier, France*
E-mails: felipe.vargas-rojas@ird.fr, vincent.armant@ird.fr, isabelle.mougenot@umontpellier.fr

Abstract.
Global forests are critical for carbon sequestration and to face climate change challenges. Forests are distributed across the planet and both local and international organizations have led efforts to develop information systems to store and manage forestry data. Such data includes spatio-temporal observations, metadata about tree species and their traits, for example the trunk diameter at breast height. Forestry data is in nature heterogeneous, multi-scale, and multi-source. In practice, however, forest data often remains stored in local installations and is rarely shared as open data, thereby neglecting the Findable, Accessible, Interoperable, and Reusable (FAIR) principles. The Semantic Web community has contributed several standards in related areas such as: phenotypic descriptions of species (OBO/PATO), spatial objects (GeoSPARQL), observations and measurements (SOSA), units of measurements (QUDT), among others. However, to our knowledge, there is a lack of comprehensive studies demonstrating how these standards can be arranged and combined to facilitate the curation, reuse, and exploitation of forestry datasets. To cope this gap, we propose the OneForest Knowledge Base (OneForestKB) that is based on a semantic profile and novel methods to validate and enrich forestry datasets. We follow a strategy of reusing existing ontologies as the definition of a semantic profile suggests. The relevance of this strategy is demonstrated through two use cases in the Amazonian forest of French Guiana, showing how to improve data quality through spatial validation rules based on the W3C constraint language, SHACL, combined with the OGC standard GeoSPARQL; the SHACL rules are generalised and shared to be applicable to any system using the GeoSPARQL model. Besides, this work provides a set of data enrichment rules relevant to forestry studies, which enable to enrich geographic regions with the calculation of ecological indexes as proxies of biodiversity. OneForestKB is designed to be extensible, allowing new validations and inferences to be added based on specific use cases.

Keywords:
Forest, Knowledge Graph, Spatial Modelling, SHACL, GeoSPARQL

1. Introduction

Forests play a critical role in the carbon dioxide absorption cycle. However, due to the extreme climate conditions caused by climate change, forest resilience is in danger [7]. Recent technological advances in areas such as remote sensing and the Internet of Things (IoT) have yielded an enormous growth of data acquired from multiple sources (satellites, sensors, surveys, etc). Forestry datasets are a kind of ecological terrestrial data where trees and land

*Corresponding author. E-mail: felipe.vargas-rojas@ird.fr.

areas are the main entities; they are in nature multi-scale, multi-source, and disparate [46]. A correct and timely understanding of forest interactions may facilitate forest protection and decision-making activities. For instance, to know what species of trees are disappearing in a given land area, or if an insect or disease causes disturbances in the forest balance.

Governments and institutions have led a vast effort to organise and store forestry data in information systems, which include forest inventories and land cover maps. However, those systems are poorly interoperable and the collected data remains in silos for particular use cases [44]. Therefore, combining multiple datasets can be particularly challenging; for example, data can be in diverse formats [24]. Multiple studies and platforms focus on sharing and managing forestry data. The BASIFOR [11, 41] platform aims to organise the evolving Spanish forest inventory, but this platform has shown some difficulties in addressing data integration issues [24]. Another example of a large database related to forestry is the Database of European Forest Insect and Disease Disturbances (DEFID2). This database contains over 650,000 georeferenced records corresponding to eight countries; gathered data covers incidents over a period from 1963 to 2022 [21].

The Semantic Web (SW) [9] has gained momentum in research areas related to forestry, including ecology, agriculture, geospatial, etc. SW provides several standards and models that have been adopted and validated in multiple domains. SW facilitates the construction of Knowledge Graphs (KGs). KGs are concrete data structured in the form of labelled graphs conforming to SW standards. KG's schema is usually described using ontological knowledge that concerns entities and relationships describing a particular domain.

Due to the multidisciplinary nature of forestry datasets, studies addressing diverse aspects such as spatiotemporal, biological, etc..., are also relevant to consider in this work. Different KGs have been constructed recently for diverse applications: a KG for geologic mapping in Korea [32], a KG for seismic representation and prediction [17], a KG for climate modelling referred as the LinkClimate [58] platform, a KG for Water healthy [37], and for the particular use case of forest fire [14]. More and more, research domains are figuring out that traditional databases are insufficient to address the intrinsic complexity of their datasets.

The SW technologies and KGs have been demonstrated to be adequate to address disparate and multi-schema data in related environmental works. However, to our knowledge, there is currently a lack of a common application profile that leverages existing referential ontologies intended particularly for forestry datasets. An application profile aims to extract only the relevant terms from diverse vocabularies to model a particular context [48]. We know of a related application profile denominated TERN [49], which models ecological entities, but it ignores certain aspects of the forestry datasets. For instance, what is a tree?, what species of trees are there?, what is a forest?, how to model ecosystem services?. Ontologies and vocabularies such as Darwin Core, GEMET, ENVO, along with others are adequate to model those aspects. In that regard, we propose a more specialised profile extracting from the mentioned resources the particular terms and relationships that can represent forest datasets. Besides, in this work, we go beyond that; we leverage SW technologies to exploit the structured data. In particular, we demonstrate how the use of SW geospatial services can enable data validations and completion by merging two main SW technologies: the GeoSPARQL [42] framework and the Shapes Constraint Language (SHACL) [36]. SHACL has gained attention concerning the exploitation of graph data [51, 60], but further research should be conducted to systematically address concrete use cases.

This work is carried out within the EU-China project, *eco2adapt*[1]. This project brings together 16 Chinese and European expert organizations in the fields of forestry, ecology, and climate change, along with 19 affiliated entities, from 11 countries. Many cross-regional, multi-scale research works focus on indicators acquisition and resilience evaluation. The project joins interdisciplinary studies while strengthening the international community's consensus on forestry coping with climate change.

The contribution of this work are presented as follow:

- An semantic profile for global forestry data exploiting referential ontologies such as SOSA , GeoSPARQL and Time.
- A web user interface enabling one to navigate and visualise forestry data.
- Two use cases enabling (i) spatial validations and (ii) data enrichment from trees' species facts.

[1] https://www.eco2adapt.eu/

– A set of SHACL shapes enabling to validate GeoSPARQL predicates against concrete geometries.

This work is organised as follow: Section 3 details the background and introduces the knowledge base OneForestKB. Besides, Section 4 explains the prototype, including the web application and the triple store server. We showcase OneForestKB with a real-world study case in Section 5. Finally, Section 7 exhibits the conclusions and perspectives.

2. Related Work

This section details four Semantic-Web studies related to forest data, we begin describing the different approaches and then we provide a detailed comparison of them.

The Cross-Forest European project [24] leverages SW technologies in order to create ontologies and web tools to facilitate data integration and visualisation. Their focus area are forest inventories from Spain and Portugal, following a bottom-up approach by creating new terms and ontologies to represent new datasets. In particular, they develop the Forest Explorer [52] platform, which is a tool to access Cross-Forest[2] datasets. In other work, the authors propose the FooDS project [27] including a novel ontology and knowledge graph to describe data from forest observatories, the project focus on wild life interactions in forest. However, we argue that the existing standards already cover numerous aspects of forestry data. A recent study introduces the SORSOnt ontology [2] aiming to model deforestation context based on remote sensing indicators. They adopt the knowledge provide by the weather to enrich and improve deep learning algorithms. Finally, the TERN [49] semantic profile represents an Australian initiative to standardise plot-based field surveys. Although it is a mature, long-term project and the most closely related approach, the ontology omits several critical concepts essential for forestry datasets, as detailed later in the comparison.

In general, the mentioned studies give insufficient attention to the particular concepts and relationships necessary in forest studies, the referential ontologies employed, and how to leverage Semantic Web services to improve aspects of forestry datasets, such as data quality and enrichment.

Table 1 compares the above-mentioned approaches to model terms related to the forestry domain. We also include OneForestKB in order to position our profile relative to the others. It is important to note that each work focuses on different areas, and their design decisions reflect their respective goals. Given these differing purposes, we present this comparison to provide a clearer understanding of OneForestKB. In the table, we present the types of **approaches**, distinguishing between semantic profiles and ontologies. This distinction is based on the observation that some approaches introduce several new concepts and relationships, whereas we made an effort to identify appropriate terms within existing resources instead. For instance, FooDS defines a new term, *foo:Observation*, and uses *rdfs:subClassOf* to map it to the SOSA ontology. Conversely, OneForestKB directly utilizes existing ontology terms and extracts only the specifically needed relationships. Concerning the biological dimension of **tree species**, neither TERN nor FooDS provide a standard model for this aspect. It is important to note that their focus areas are field surveys and wildlife interactions, where tree species are less relevant to their modelling and use cases. **Measures** are considered in all the studied approaches; however, for OneForestKB and TERN, this dimension is central and serves as the starting point for aligning the other dimensions. All the approaches address the **geospatial dimension**, with some using GeoSPARQL and others employing W3C standards such as Basic Geo (WGS84 lat/-long)[3]. In our approach, since we exploit geospatial validations, a more comprehensive model such as GeoSPARQL is more appropriate than Basic Geo. The **environmental dimension** is particularly important for forestry experiments, as regions and environmental entities facilitate the description of land use. Notably, the term "forest" is well-defined within the OBO Foundry's ENVO ontology (obo:ENVO_00000111, forested area). The **time dimension** is disregard in the Cross-Forest platform due to their bottom-up strategy from which a ontology is defined for each type of dataset and time plays a secondary role in their use cases. **Ecosystem services** are crucial for studying forests, as they enable the description of the main uses of this natural resource across multiple activities, such as

[2] a LOD resource: https://crossnature.eu/data/
[3] https://www.w3.org/2003/01/geo/

Table 1
Comparison of Semantic Web approaches for representing forestry-related dimensions. (*) SH=SHACL, RS= Remote Sensing, GEO = GeoSPARQL

	TERN	Cross-forest	FooDs	SORSOnt	OneForestKB
Focus area	field surveys	forest inventories	wildlife research in forest	RS* & deforestation	forest resilience & living labs
Approach	semantic profile	ontology	ontology	semantic profile	semantic profile
Conceptualisation					
Tree species		√			√
Measures	√	√	√	√	√
Geospatial	√	√	√		√
Environment					√
Weather				√	
Time	√		√	√	√
Ecosystem services					√
License					√
Derived data	√				√
Features & Services					
User interface		√			√
Species alignments		√			
Data validation	√ (SH*)				√ (GEO-SH*)
Data enrichment					√ (numerical)

cultural (recreation, education), provisioning (timber production), and regulating (heat protection) services. Regarding multi-source data, having a clear **licensing** framework is also important when multiple institutions contribute as data providers. Finally, **derived data** refers to data produced by running a procedure. By using SOSA as a central component, the notion of a used procedure enables modeling of such derived data; both TERN and OneForestKB employ this modeling strategy; although extensions are required to have a complete description of derived values. Note that this comparison concerns the presence or absence of particular dimensions, but not the level of description or quality.

In addition to conceptualization, we compare features and services. A **user interface** is an essential service for exploring the data, which is particularly addressed by Cross-Forest through a platform offering multiple features. The TERN profile does not specify a particular interface and focuses instead on terms, ontologies, and mappings. FooDS provides multiple use cases and collaborative notebooks for data visualization; however, a dedicated user interface is not yet available in the current version. Our work contributes both a user interface and multiple APIs.We identified an interesting feature in the Cross-Forest platform, incorporated into this comparison: **species alignments**, which are relevant for forestry studies. Cross-Forest provides useful species alignments to Wikidata and the Darwin Core Vocabulary. The other approaches do not include this functionality, and we plan to add such a module in future versions. Data validation is less emphasized in the reviewed studies. In the latest version of TERN (2025-06-03), a set of SHACL shapes is included as part of the specification. Our approach addresses validation with a focus on geospatial relationships, demonstrating the potential of Semantic Web standards for data curation. Data enrichment is only explicitly addressed in our work, where we use existential rules to derive new information from mathematical computations and aggregations. This module will be described in detail later. It is worth noting that data enrichment can also be achieved through ontological reasoning; however, to our knowledge, none of the presented approaches explicitly detail this process.

To summarize, we examined various Semantic Web approaches related to forest data, noting that each study focused on different areas. Rather than claiming superiority for our approach, we aim to demonstrate that similar design decisions were made and comparable dimensions and concepts were considered across the works. The

core principle of standardization is to reuse existing resources—for instance, we could adopt species mappings from Cross-Forest or SHACL shapes proposed in TERN, while these studies could leverage our geospatial validation or enrichment rules. Since forestry is an experimental science spanning multiple dimensions, we view these works—and ours—as part of a broader community effort toward standardisation.

3. Materials and Methods

3.1. Background: Semantic Web standards

OneForestKB builds upon the W3C consortium web standards. Notably, we primarily use the Resource Description Framework (RDF) [35], RDF Schema (RDFS) [40] and the Web Ontology Language (OWL) [62] to structure the data and represent domain rules, as well as SPARQL [42] to access the semantically enriched data. While RDF and RDFS offer a very general framework for knowledge representation, a number of functionalities are still lacking to provide the required level of expressiveness. When it comes to describing complex situations, a large number of applications make use of large ontologies containing complex entities that need to be effectively described. These shortcomings have led the W3C to propose a standard for ontology description on the Web, OWL (Ontology Web Language) which is built from the RDF model (RDF-Based Semantics) and is influenced by description logics (Direct Semantics) [61]. Finally, SPARQL (SPARQL Protocol and RDF Query Language) is the W3C standard language for querying and manipulating RDF graph triples.

OneForestKB is presented as a Knowledge Base. In this work, we consider a knowledge base referring to the definition given by Ontotext[4]: "The KB is a collection of interlinked descriptions of entities (real-world objects, events, situations, or concepts) that allows this knowledge to be stored, analysed, and reused in a machine-interpretable way". We adapt this definition to organize interrelated descriptions dealing with key entities in forest ecology. We also consider that a knowledge base corresponds to two sets, namely, a set of concepts and relations between them and a set of facts that conform to the preceding concepts and their relations. As a result, the knowledge base also corresponds to an ontology expressed in OWL 2 [26], which is made up of a terminological component or TBox (schema), and an assertion box or ABox (data). The notion of ontology originated in philosophy and means "theory of being". In computer science, the definition given by [50] defines ontology as a shared understanding of a domain of interest. An ontology defines explicitly, consensually and formally the terms used to describe and represent a field of knowledge. We will use the terms knowledge base and knowledge graph interchangeably to refer to the annotated data, whereas the terms semantic profile and ontology to refer to the schema level information. A knowledge graph [30] is approached as a fairly loose structure that applies a graph-based abstraction to many interrelated entities that may come from different sources and are mobilised in a timely manner in different analysis tasks.

SHACL. In order to address particular use cases, we rely on the Shape Constraint Language (SHACL) [36], which has been a W3C recommendation since 2016. Whilst RDFS and OWL offer a rich expressivity to model flexible structures, their modelling is based on the Open World Assumption (OWA), which means that unknown facts are assumed to be "partially true". This behaviour is undesired in the application level [60], where calculations should assume that the existing data is complete. Technologies such as SPARQL and SHACL are Close World Assumption (CWA) thus unknown facts are assumed to be "false". CWA is more suitable for calculations [51] and validations [60].

SHACL enables to validate constraints against a knowledge graph. Formally, following the abstract syntax defined by Corman et al. [16], a SHACL schema $\mathcal{S}$ is represented as a triple $\langle S, targ, def \rangle$ where S is a set of URIs referred as shape names, $targ$ is a function that assigns a target query for each shape name $s \in S$, and def is a function that assigns a conjunction of constraints for each shape name $s \in S$. For simplicity, $targ(s)$ is usually about an RDF class, that can be evaluated against a given RDF graph G. This query returns a set of URIs called *target nodes*. Note that we make use of the term *focus node* to refer to the specific *target node* taking part during the validation step. The constraints in $def(s)$ are described following this grammar:

[4] https://www.ontotext.com/knowledgehub/fundamentals/what-is-a-knowledge-base/

$$\phi \quad ::= \quad \top \mid s \mid I \mid \phi \wedge \phi \mid \neg\phi \mid \geqslant_n r.\phi \mid \mathrm{EQ}(r_1, r_2) \tag{1}$$

The s refers to a shape name, the I to a valid IRI, r is a SHACL path, and $n \in \mathbb{N}^+$. For more details about the grammar refer to [15].

Considering the definition of a SHACL schema, a SHACL shape consists of three elements: $\langle s, targ(s), def(s)\rangle$. Example 1 illustrates a shape named *:ObsShape* having as target a SPARQL query that returns all the instances of the class *sosa:Observation* (a.k.a. the target nodes). $def(:ObsShape)$ contains two constraints stating that an instance of the class *sosa:Observation* must have exactly one *sosa:hasFeatureOfInterest* and one *sosa:hasObservedProperty* property. If an instance has twice those property or the property is not present the SHACL reports that the RDF graph is invalid against this shape.

Example 1. *A SHACL shape for the class sosa:Observation named :ObsShape*

$$targ(:ObsShape) = \textbf{\textit{SELECT}}\ ?x\ \textbf{\textit{WHERE}}\ \{?x\ a\ sosa{:}Observation\}$$
$$def(:ObsShape) = (=_1\ sosa{:}hasFeatureOfInterest.\top) \wedge (=_1\ sosa{:}hasObservedProperty.\top)$$

Beyond constraints, SHACL also enables Advanced Features (unofficially, not yet included in the standard) such as different kinds of rules. We exploit those features in this work. For instance, we may extend the formalisation with a new function ϵ as follow:

Example 2. *A SHACL enrichment rule for creating the property :day for all the observation's instances. The property is derived from another property containing a full date information (sosa:resultTime). The semantic of BIND and DAY are the same as in the SPARQL formalisation.*

$$\epsilon(:ObsShape) = \{e_1 : sosa{:}resultTime(sh{:}this, ?date) \wedge BIND(?day, DAY(?date)) \rightarrow :day(sh{:}this, ?day)\}$$

3.2. Key Modelling Decisions

We assembled a set of framework ontologies[5] that can serve as a flexible schema to model all data and knowledge collected within Living Labs (LLs). LLs are open innovation ecosystems in real-life and experimental environments fostering co-creation among the main actors of the Quadruple Helix Model, namely: Citizens, Government, Industry and Academia [20]. The European Network of Living Labs (ENoLL) is the association that certifies and organises LLs in Europe. In the context of this study, LLs provide in-situ monitoring of forested areas. Given the wide variety of data available, here we detail certain key decisions that have enabled to model forest aspects in a generic and flexible manner. More precisely, we attempt to provide an ontological model sufficiently flexible to adapt to all the observations that may be collected in multiple LLs. Fortunately, different public framework ontologies are available on the web, which are suitable to express the diverse dimensions we expect to support in our KB.

OneForestKB's primary concept is an "Observation" that occurs in a time, a particular space, and measures a property observed from a feature of interest. LLs data can be modelled exploiting this observational paradigm, which adapts to address several domains as shown from implementation such as the Extensible Observation Ontology (OBOE) [38]; the Sensor, Observation, Sample, and Actuator ontology (SOSA); and the I-ADOPT [39] framework. For our modelling, we prefer SOSA because of their simplicity and multidisciplinary adoption. Concretely, OneForestKB main block ontologies include:

- SOSA ontology [34] for describing sensors, observations, samples and actuators.
- GeoSPARQL standard [42] for representing and querying spatially linked data.
- OWL-Time ontology [28] for describing temporal concepts.

[5] Framework ontologies or upper domain ontologies are dedicated to describing high-level concepts for a given theme and serve as a basis for integrating and interconnecting more specific ontologies known as domain ontologies

SOSA: Sensor, Observation, Sample, and Actuator Ontology

QUDT: Quantities, Units, Dimensions, and Types Ontology. We use quantities to annotate observations such as the tree diameter and height

ENVO: The Environment Ontology. The environmental definition of a forest is given for the resource obo:ENVO_01001243

DWC: Darwin Core Vocabulary. Terms for annotating species, genus and family of threes. A tree can be modelled as a dwc:Organism

GEMET: the GEneral Multilingual Environmental Thesaurus. A tree is defined in gemet:8664

SDGIO: Sustainable Development Goals Interface Ontology (SDGIO). We extrac from this ontology notions about ecosystem services, including regulation, cultural, and provisioning ecosystem services

PTO: Plant Trait Ontology. We use traits such as the stem diameter (obo:TO_0020083)

Table 2
Overall view of the OneForestKB profile highlithing main ontologies and vocabularies

GeoSPARQL, OWL-Time and SSN (Semantic Sensor Network ontology) including SOSA have been elevated to the status of standards by either the Open Geospatial Consortium (OGC) or the World Wide Web Consortium (W3C), or both, making all three particularly interoperable. The adoption of standardization from the beginning makes the OneForestKB interoperable, easily extensible, and durable, also facilitating community development. The idea of assembling these standards is not novel, as shown in the TERN ontology [49]. An overview of the most relevant ontologies and vocabularies that compounds the profile is shared in Table 2.

3.3. Semantic Profile for OneForestKB

The knowledge model used to structure the OneForestKB does not introduce any new modelling elements. We consider that there are sufficient semantic components from which we can assemble the main concepts and relationships without having to define new ones. Our approach is called 'mix and match'. This is why we speak of a semantic profile rather than a knowledge model, since the conceptualisation effort is focused more on integrating existing elements.

We select the elements and relationships of interest in the knowledge models selected, and we relate them. For each object of interest, we organise its characteristic elements into observations that can be temporised and spatialized. The GeoSPARQL (spatial dimension), OWL-Time (temporal dimension) and SOSA (observation paradigm) ontologies will be central to the approach, and are in fact very present in the general class diagram.

Complementary to the ontologies mentioned above, we incorporate more domain-specific ontologies such as the provided from the OBO sphere. For example, Plant Ontology (PO) [12], Environment Ontology (ENVO) [13] or Plant Trait Ontology (TO) [4] and Phenotype And Trait Ontology (PATO) [25], whose classes will refine the concept of observable property, or Creative Commons ontology (CC Rel)[6][1] or PROVO[7] for everything to do with property rights protecting data authors. We have also adopted the DWC (Darwin Core) [55] standard for everything relating to forest biodiversity, and in particular for the taxonomic identification of the trees populating the forest. We also use metadata standards such as Dublin Core Terms vocabulary [6] to enrich datasets with additional information.

The semantic profile will play a key role in the organisation of the knowledge base, as it will not only establish the organisation within the knowledge base, but will also provide several complementary instances of the OneForestKB knowledge base that can be deployed at the various project partners' sites and consulted together via federated queries. The profile will also facilitate the construction of external services such as consultation services or statistical data processing.

The semantic profile (see Figure 1) integrates classes of various ontological components. The colour code has been retained so that the origin of each class can be seen quickly. The stereotypes also indicate the origin of each

[6]https://opensource.creativecommons.org/ccrel/
[7]https://www.w3.org/TR/prov-o/

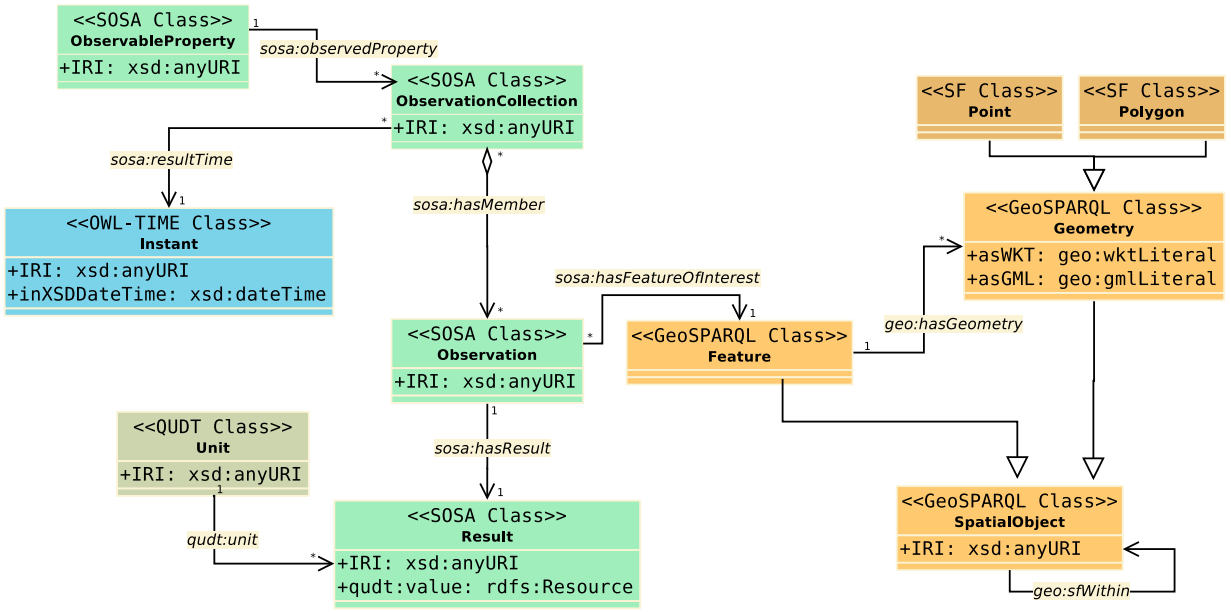


Fig. 1. UML class diagram representing OneForestKB's profile

class. For example, the Observation class is labelled with the "SOSA Class" stereotype. Properties are indicated with their namespace. For example, the object property *sosa:hasFeatureOfInterest* is used to link an observation to an observed object, which in this case has a spatial dimension since it is of type with the GeoSPARQL Feature class. To make the model more concise, the observations are grouped together in observation collections, which will all relate to the same observed property and which will be temporised. The QUDT ontology (Quantities, Units, Dimensions and Data Types Ontologies) [29] is used to value and type the results of the observations. The extraction of the required entities and relationships is shared in our repository [8].

For profile development, we used the ROBOT tool [33], which enables numerous command-line routines for ontology manipulation and profile generation. Once the profile was generated, we evaluated its FAIRness using the FOOPs [23] website. FOOPs is a tool for assessing compliance with the Findable, Accessible, Interoperable, and Reusable (FAIR) principles [56]. The evaluation yielded a score of 74% due to missing labels for some imported terms in the source ontology.

Main entities and relationships. The main entities in the forest domain are trees, in OneForestKB a tree is an instance of different classes, each of them addressing a particular aspect, therefore a tree can be of type:

- **sosa:FeatureOfInterest**: trees are features of interest considering the SOSA ontology, this SOSA class enables to model sensor observations about a particular tree also considering the temporal aspects of such measurements.
- **geo:Feature**: trees are features considering the GeoSPARQL specification, this classification permit to add annotation about the spatial aspects, in OneForestKB trees are geometrical points.
- **dwc:Organism**: trees are vegetal organisms; this Darwin Core term enables this biological description, which we extend using the GEMET thesaurus term for trees (gemet:8664). We employ the predicate dct:subject from the DCMI Metadata Terms (DCT) to stablish such categorical description. Note that to avoid inconsistencies, it is unsuitable to use vocabulary terms as RDFS/OWL classes.

Trees are located at specific geographical points and are distributed across various spatially delimited regions at multiple scales. In the examined use cases, we found that trees can be situated within plots, which in turn are con-

[8] https://github.com/felipe-vargas-rojas/OneForestKB

tained within living labs, while living labs are nested within broader forest ecosystems and national boundaries. The hierarchical relationship of regions within regions can be effectively modelled using GeoSPARQL functionalities. Consequently, plots, living labs, forests, and countries are represented as Features within the GeoSPARQL framework, allowing for the seamless integration of spatial data.

On the other hand, trees belong to a particular tree species. In order to model tree species we utilise the relationship *dwc:taxonID*. Whilst it is more natural to model a tree as an instance of a particular species, that information is not always known a priori. Therefore, we adopt a modelling where the species is annotated with the taxon ID when known and with a blank node when unknown. Notice that for some trees, we could have partial information, such as the genus and the family, that consideration provides more elements to justify our modelling decision.

Listing 1: An observation and a tree of interest along with a resulting measurement

```
guyafor:ob1 a sosa: Observation ;
      sosa: hasFeatureOfInterest guyafor:tree1 ;
      sosa: hasResult guyafor:rs1.
# result information
guyafor:rs1 a sosa: Result ;
      qudt: numericValue "70.05751801"^^ xsd:double.
# observation collection info
guyafor:oc1 sosa:ObservationCollection ;
      cc:license cc:ShareAlike ;
      dct:creator "ONF – Guyane " ;
      qudt:unit unit:CentiM ;
      sosa:hasMember guyafor:ob1 ;
      sosa:observedProperty obo:TO_0020083 ;
      sosa: resultTime "Tue Mar 22 00:00:00 CET 2005
            "^^ xsd:dateTime.
# Plant Trait Ontology (PTO)
obo:TO_0020083 rdfs:label "stem diameter"
```

Listing 2: Tree metadata annotated with the DWC Vocabulary

```
guyafor:tree1 a dwc:Organism, geo: Feature;
      dct:subject gemet:8664;
      dwc:family "Apocynaceae" ;
      dwc:genus "Macoubea" ;
      dwc:specificEpithet "guianensis" ;
      geo:hasGeometry guyafor:dbcx72pyngvf_g ;
      geo:sfWithin guyafor:BAFOG_IV .
# geometry information
guyafor:dbcx72pyngvf_g a sf:Point ;
      geo:asWKT "POINT( −53.98686218
            5.494214535)"^^ geo:wktLiteral .
# geometry region
guyafor:BAFOG_IV geo:hasGeometry [
      geo:asWKT "POLYGON((−53.9883281304637
            5.49479749123131, −53.986580714517
            5.49524337205015, −53.9861357530496
            5.49349236779835, −53.987883163426
            5.49304649142441, −53.9883281304637
            5.49479749123131))"
]
```

3.4. Related Examples

To begin with, Listing 1 shows triples (in the RDF Turtle format) concerning parts of the profile entities. Initially, an observation (*guyafor:ob1*) is declared as having as a feature of interest a given tree (*guyafor:tree1*). The observation is linked with a result through the property *sosa:hasResult*. Such a result is declared as a QUDT numeric value, in this example with the value of 70.0575. Notice that in the observation, we do not declare the unit of measurement or the observed property. Some attributes are redundant between SOSA observations; in a manner of factorisation, the class *sosa:ObservationCollection* enables the grouping of properties that affect numerous observations. In this listing, the given observation collection has as member the illustrated observation (*guyafor:obs1*) but also contains some predicates such as the unit of measurement (*unit:CentiM)*, the observed property (“stem diameter”), the result time as well as license and permission metadata. If another observation concerns these attributes, we should only add it as a member of the collection.

Moreover, Listing 2 displays metadata concerning aspects of a tree. We use the Darwin Core ontology to annotate taxonomic facts about trees. Since that metadata can be incomplete, we add to the tree all the known annotations, including the family, the genus, and the species. Secondly, in this example, we add a spatial feature to the tree. The example shows a tree modelled as a *sf:Point* as well as a geometry fact declaring that this tree is within a region named *guyafor:BAFOG_IV*. That region is also a geometry but a polygon instead of a point. One validation question

about the within predicate can be to confirm whether or not this point is effectively within that polygon. We address that kind of validation in the use cases.

3.5. OneForestKB's Services

OneForestKB was built upon well-established open web standards in related fields. Here, we further describe how to exploit this knowledge base for real use cases. Thus, providing a more useful resource for the forestry-domain users. We explain that in an abstract manner and we revisit some of these services in the use cases later.

Quality Assurance. Graph-based data can be validated with a range of technologies including the W3C recommendation SHACL. Indeed, validating the spatial aspects of these datasets is of interest to forestry studies

Data Linking. Having the forestry data in a linked format such as RDF, can help to discover new facts by linking the local data with external databases such as Wikidata [53] or GBIF [47].

Inference Services. Different Semantic Web standards enable inference services to derive new facts, most traditional ones include OWL axioms. However, rules languages such as SHACL rules are more expressive to model particular use cases in a declarative and FAIR fashion.

4. OneForestKB Prototyping

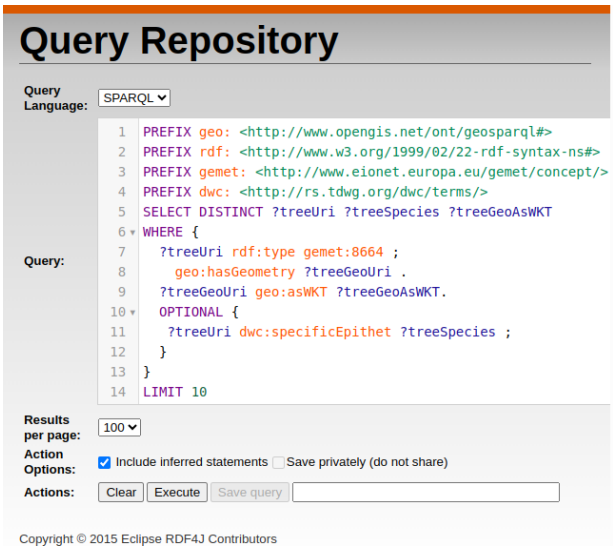


Fig. 2. SPARQL Query through the RDF4J Workbench

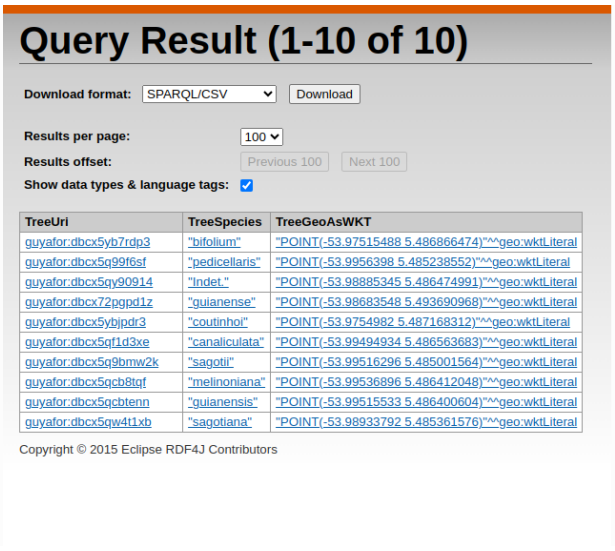


Fig. 3. SPARQL Query results through the RDF4J Workbench

4.1. The RDF server

We have deployed an initial web-based solution, which is currently intended as a prototype. This solution includes a knowledge base instance containing all the data from some living labs of the project, the information is hosted on an RDF4J triplestore (146,230 triples). Eclipse RDF4J[9] (formerly OpenRDF Sesame) is an open source framework

[9] See https://rdf4j.org/

designed to host and manipulate data in RDF format. Its origins lie in the IST "On-To-Knowledge" project (1999 to 2002), so it seemed natural to us to use this framework as part of a new European project. RDF4J is based on Java and incorporates an RDF triplet persistence solution, which we use first and foremost. Other RDF persistence systems [3], such as Apache Jena Fuseki [22] or Virtuoso [45], could also have been used. We use Amazon Web Services to make the RDF4J server (RDF database server) and RDF4J Workbench (servlet container based on the Apache Tomcat engine) available. The RDF4J server provides access to RDF4J repositories via SPARQL endpoints. We present screenshots (Figures 2 and 3) illustrating the SPARQL access point to the knowledge base. The elementary query, showing ten individual trees and their genus, is taken as an example.

The SPARQL query is submitted to the system via the dedicated input zone in Figure 2. We have given a very simple query as a first example, but we could also propose more complex queries of interest to ecologists. As a first example, we illustrate a query that requests the URI, species, and geometry of the first ten trees. The result of the SELECT query is displayed in tabular format in Figure 3. Various export formats (XML or JSON) are also available.

4.2. Web Application

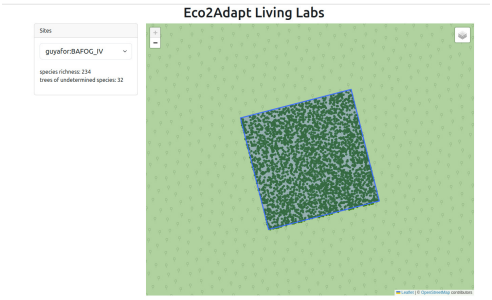


Fig. 4. Details of the BAFOG_IV parcel via the interface

Fig. 5. A selected tree within the plot

We developed a web user interface, which is available at https://purl.org/oneforestkb-bafog/app and will be expanded as project partners express their needs. For the moment, we are displaying the spatialised objects (site, plots and trees) over an OpenStreetMap (OSM) geographical map [3]. The work takes advantage of SPARQL queries on OneForestKB and Javascript developments. The React framework[10] and the OpenLayers library [11] are also being used. The Figure 4 shows the 4-hectare plot from the living lab guyafor named BAFOG_IV, with the green dots corresponding to the trees located within this plot. The interface allows you to position yourself at different spatial scales and select a plot of interest. The Figure 5 allows you to select a tree and view its taxonomic information, in particular its family, genus and species. Here, it is shown the tree with the identifier *guyafor:dbcx72pvtgbf*, of the species *Lecythis idatimon*.

[10] a Javascript framework for building user interfaces https://react.dev/
[11] OpenLayers a JavaScript library for displaying map data on the web

Path	Description	Type	Parameters	Output
/livinglab/validate/sfWithin	Validate the geo-predicate sfWithin against the concrete geometries	GET	GeoFeatureURI	SHACL Report
/livinglab/validate/sfEquals	Validate the geo-predicate sfEquals against the concrete geometries	GET	GeoFeatureURI	SHACL Report
/livinglab/validate/sfIntersects	Validate the geo-predicate sfIntersects against the concrete geometries	GET	GeoFeatureURI	SHACL Report
/livinglab/inference/totalTrees	Infer the total number of trees	GET	GeoFeatureURI	RDF graph
/livinglab/inference/AbsoluteAbundanceIndex	Infer the Absolute Abundance Index	GET	GeoFeatureURI	RDF graph
/livinglab/inference/RelativeAbundanceIndex	Infer the Relative Abundance Index	GET	GeoFeatureURI	RDF graph

Table 3

OneForestKB' geospatial services. The GeoFeatureURI parameter provides the delimited area and is used to build the subgraph G'. Validation services return a simplified SHACL report, whereas inference services provide the inferred graph, after applying the rules, in the JSON-LD format.

4.3. API REST for OneForestKB' Geospatial Services

Accessing OneForestKB is also possible through an API. This API can be consumed from different sources such as programming scripts, including Jupyter notebooks. The API offers general access to the most common entities, such as living labs, sites and tree information. Moreover, the API permits access to the OneForestKB' geospatial services, these services illustrate the exploitation of the Knowledge Graph data based on Semantic Web standards. For this version, all the provided services are based on the Advanced Features of SHACL (SHACL-AF), main types of services include validation and inference (see Table 3). Additionally, the users can define parameters to guide the service for particular purposes. One example of a service is the one validating the GeoSPARQL predicate *geo:sfWithin* against real geometries. Notice that the geo-predicates can be used even if the geometries are incoherent, this service confirm if the geometries also respect the within fact. Whilst this service can be applied to the full graph, a user can delimit the area by adding a parameter to the service call (e.g., the experimental site *guyafor:BAFOG_IV*). As a result, the service will only be applied in entities affected by that particular area. To achieve that behaviour we define a referential workflow that handles the user parameters:

1. Considering a particular geo-feature entry by the user (e.g., *guyafor:BAFOG_IV*), construct a new subgraph (G') with the related spatial objects and their properties. The construct query is shared in Appendix B.
2. Perform the validation or inference rules in the new subgraph G'.

This basic workflow helps to optimise the rule execution by avoiding unnecessary calculations and storage. The current version has a particular focus on geo-features and geo-predicates but can be extended for different projections of the data graph. Similarly, the inference services also are applicable in a delimited area, in contrast to the validation services, the inference ones produce new derived data to enrich the data graph. The Table 4 lists three inference services with their description. These services will be invoked later in the use cases.

One common use of OneForestKB is within Jupyter Notebook documents, considering that researchers in multiple eco-environmental disciplines are familiar with this tool. OneForestKB can be accessed via SPARQL queries using libraries such as *rdflib* but also through the REST API. Whilst the SPARQL endpoint is more flexible, the REST API simplifies the access and provides more prepared data. As a result, the obtained outputs can be analysed and displayed using visualisation libraries including *matplotlib*.

5. Case Study: Amazonian Forest in French Guiana

We showcase through a real-world dataset coming from the Amazonian forest in the French Guiana that our model is useful to model trees, geographic regions and measurements such as the tree's diameter. We also exploited the annotated data in two use cases having different purposes, one for validation and the other for data enrichment.

Site	# of species	# of trees	# unknown species
BAFOG_I	13	3744	7
BAFOG_II	16	3271	4
BAFOG_III	14	4098	6
BAFOG_IV	18	3429	2

Table 4
Dataset description for the living lab *BAFOG* and their multiple sites

Dataset description. For this study, we retrieved information from two sources concerning a French Guiana living lab. The source datasets are available online. The observed measurements were obtained from a dataverse [12]. This dataset was described in tabular format, which is why we produced scripts to transform this data into our target profile. To add more contextual information, a second public source [13] was requested, in particular to feed the geospatial information about the studied plots and sites. The Guiana's living lab is identified in the KB with the URI *guyafor:BAFOG* along with four different sites (more specific regions): *guyafor:BAFOG_I*, *guyafor:BAFOG_II*, *guyafor:BAFOG_III*, *guyafor:BAFOG_IV*. For each site we describe the number of distinct species and trees in the Table 4. Additionally, this dataset contains time-stamped information about the diameter of trees.

5.1. Use Case I: Validation of Spatial Predicates

OneForestKB contains spatial annotations using the GeoSPARQL ontology. GeoSPARQL provides (i) spatial predicates and (ii) spatial functions, in this evaluation we demonstrate that the datasets may contains incorrect facts. For instance, a fact expressing that a tree is within a site could be validate: by confirming if their geometries are one within the other.

We implement a SHACL shape (see Listing 3) that leverages GeoSPARQL functions to validate the GeoSPARQL predicates. The idea of combining SHACL and GeoSPARQL was already explored by Debruyne et McGlinn [18]. However, our SHACL shape automatically reads the GeoSPARQL predicate and triggers the validation without adding any additional namespace or component.

In particular, this shape addresses three main issues. Firstly, capturing the triples concerning the *geo:sfWithin* predicate, for that goal the shape declares as target nodes the subjects of the *geo:sfWithin* property. Secondly, the transitive nature of this predicate. For instance, a *tree* can be within a *plot*, and the *plot* can be within a *site*. Therefore, we expect the *tree* to be within the *site* as well. For that purpose, the shape declares a property constraint concerning a special *sh:path*, where the *geo:sfWithin* predicate is expressed in a transitive manner with the *sh:zeroOrMorePath* path expression.

The *geo:sfWithin* relationship is inherently transitive, as noted by Battle and Kolas [8], although the GeoSPARQL specification assumes non-transitivity. The transitivity of topological relationships remains an open issue for specific implementations. One intuitive approach to address this is using SPARQL path expressions, as implemented in our SHACL rule. Currently, our data contains paths of only three depth levels (e.g., living lab/plot/subplot/tree), and no cyclical issues have been detected.

The path used in the rule searches for at least one path between the GeoFeature and the given tree. As noted in the SPARQL Property Paths specification[14], if multiple paths exist, only a unique solution is returned, and validation is not repeated. However, future versions should address cycles, as they are permitted and included in matches. An automatic cycle detection mechanism or safeguard for the path will be necessary; otherwise, the rule could enter an infinite loop. One solution is to define a max size for the path steps such as: *geo:sfWithin{1,10}*.

Finally, the GeoSPARQL standard suggests that the spatial predicates relate spatial objects (*geo:Geometry*, *geo:Features*). In order to cover this issue, the select clause in the Listing 4 uses advanced path expressions,

[12] https://dataverse.cirad.fr/dataverse/CIRAD/?q=guyafor
[13] https://www.guyane-sig.fr/geonetwork/srv/api/records/53ef33ab-3c26-4a1c-a8ee-e0bb328753df
[14] https://www.w3.org/TR/sparql11-property-paths/

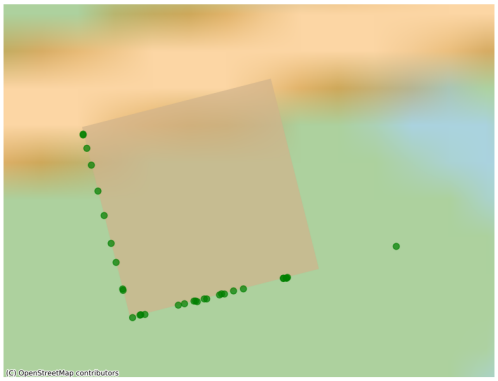
before invoking the spatial function *geof:sfWithin*. Note that the predicate *geo:sfWithin* and the spatial function *geof:sfWithin* have different namespaces and purposes. This modelling strategy could be easily adapted to model validations over other GeoSPARQL. We provide the list of validation shapes concerning the seven Simple Function predicates in a public repository: https://purl.org/geoshape.

Listing 3
A SHACL shape for validating the geo:sfWithin's predicates

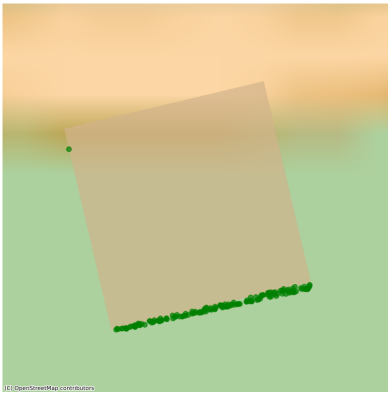
```
ex:sfWithinShape a sh:NodeShape;
        sh:targetSubjectsOf geo:sfWithin;
        sh:property [
                sh:path (geo:sfWithin [sh:zeroOrMorePath
                        geo:sfWithin]); # transitive
                # sh:path geo:sfWithin;
                sh:sparql [
            sh:message "Spatial object {$this} is not within
                    {?o}" ;
            sh:prefixes ex:, geo:, geof:;
            sh:select """<see query in Listing 4>""".]].
```

Listing 4
select clause for the SHACL shape ex:sfWithinShape

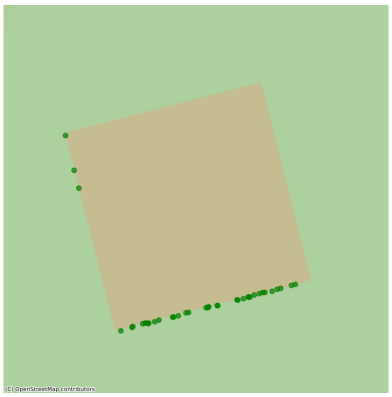
```
SELECT $this ?o
{
      $this $PATH ?o.
      $this (geo:hasGeometry/geo:asWKT|geo:asWKT) ?
            wktLiteralA.
      ?o (geo:hasGeometry/geo:asWKT|geo:asWKT) ?
            wktLiteralB.
      FILTER( ! geof:sfWithin(?wktLiteralA,?
            wktLiteralB)).
}
```



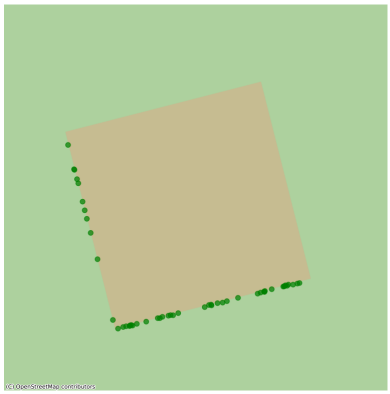
(a) BAFOG_I



(b) BAFOG_II



(c) BAFOG_III



(d) BAFOG_IV

Fig. 6. Trees found out of their corresponding site

Results. The *ex:sfWithinShape* is evaluated against the data graph. A SHACL validation report is produced indicating how many distinct trees contained incoherent geometries. Notice that the geometries can be correct in format (a valid WKT) but the validation concerns the semantics of an element to be within another. This validation was able to find 279 erroneous trees, which would be difficult for humans to detect manually. Figure 6 displays four subgraphs, each one referring to a particular site (i.e., BAFOG_I, BAFOG_II, BAFOG_III, BAFOG_IV). The background of each figure corresponds to a layer of Open Street Map (OSM). The validation was performed against each site individually. Thus, this figure allows us to visualise the trees found outside of a particular site. The majority of trees violating the validation occurred in the polygon's borders. Conversely, the subplot 6a presents a particular outlier on the right bottom. Manual validations can be insufficient to cover the thousands of trees and combinations presented in the data. Instead, this rule enabled a more robust and complete validation. For the trees in the borders, causing validation errors, we have some assumptions: (i) rounding errors in the coordinates, (ii) integration errors and inconsistencies between the site information and the trees, since they were gathered from different sources, (iii) instrument errors. Diving into the results, for BAFOG_I 32 trees violated the rule, meaning less than 1% (32 out of 3746). Concerning BAFOG_II, the rule found 158 violations, meaning almost 5% (158 out of 3272), whereas for BAFOG_III 38 errors were detected, meaning as well less than 1% (38 out of 4099). Finally, BAFOG_IV exposes 50 validation errors, meaning a 1.4% of the trees present in this site. Notice that we run the validation over sites in order to demonstrate the functionality. However, users can perform more advanced studies by testing the different plots and geometrical objects.

5.2. Use Case II: Inference of the Abundance Index

This section explains how to model a numerical indicator derived from the existing data. We seek a strategy for data exploitation based on rule languages. This approach enables us to gain in modularity and reproducibility. Graph-based calculations are reached from different points of view; for instance, the Semantic Web Rule Language (SWRL) [31] incorporates them through built-ins. Given the continuous nature of calculations, they are often ignored in the rule saturation and the automated reasoning in order to avoid undecidable scenarios. We use instead SHACL rules that are more flexible; SHACL rules are the evolution of the SPARQL Inferencing Notation (SPIN)[15] rules.

For this particular case, we expose a rule to compute a biodiversity indicator named the Absolute Abundance Index (A_a) in a geography region (geo:Feature in GeoSPARQL vocabulary): " the sum total of individuals from a given species within a given area" [19]. Similarly, we also propose to declare a second rule to calculate the Relative Abundance Index (A_r), which relies on the A_a and is expressed as a percentage relative to the total number of individuals (N). A_r permits us to add rule dependencies; this index is computed as follows:

$$A_r = \frac{A_a}{N} \times 100 \tag{2}$$

For that purpose, we require to declare three different rules: (i) one for the A_a, (ii) a second for inferring the N and (iii) a last one to derive the (A_r). We use the term inference since the declared rules enrich the KG data with new triples to add the resulting values to the geo-feature of interest, thus new rules may assume the derived data is already materialised concerning the rule dependencies. For this study, we incorporate a new name space refereed as *ecoShapes* concerning the URI *https://purl.org/ecoShapes/*. In this regards, three new rules are added:

- *ecoshape:inferTotalTrees*(r_1) computes the total number of trees in a given area.
- *ecoshape:inferAbsoluteAbundanceIndex*(r_2) computes the Absolute Abundance Index in a given area for each family of trees.
- *ecoshape:inferRelativeAbundanceIndex* (r_3) relies on the two previous rules. Their calculation is described in Equation 2.

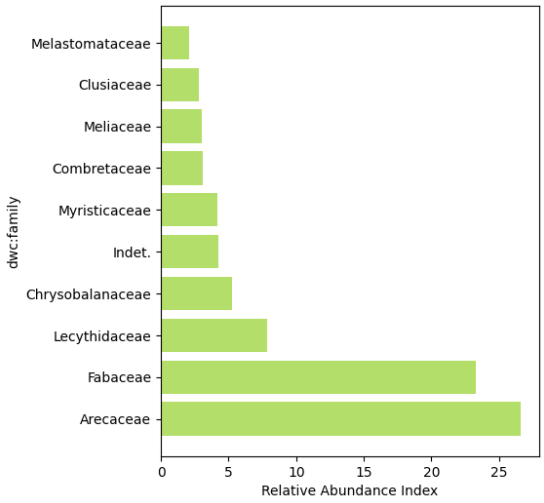
[15] https://spinrdf.org/

To illustrate, the rule ecoshape:inferTotalTrees applied to the region *guyafor:BAFOG_IV*, will enrich the data with the following fact: *guyafor:BAFOG_IV ex:hasTotalTrees 3429*, thus adding a new state of the graph. Moreover, the rule ecoshape:inferRelativeAbundanceIndex relies on the others; in that case, it should be performed on resulting graphs as follows:

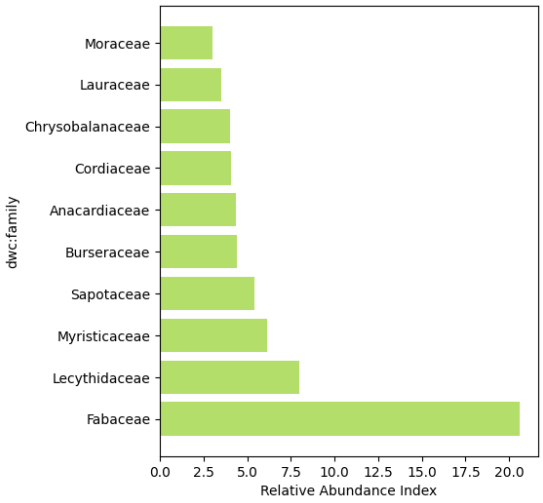
- Let G be the initial graph and $[\![r_i]\!]^{G^i}$ the evaluation of a rule r_i against a graph G^i, then,
- $[\![r_1]\!]^G \rightarrow G^1$
- $[\![r_2]\!]^G \rightarrow G^2$
- $[\![r_3]\!]^{G^1 \cup G^2} \rightarrow G^3$

Notice that r_i et r_2 are applied over the initial graph G, whilst r_3 is applied against the union of the resulting graphs of applying r_1 et r_2. The above-mentioned rules are implemented using SPARQL Construct Rules in SHACL and are shared in Appendix C. Conversely to the geoshapes, ecoshapes are more local and aligned with the OneForestKB' semantic profile.

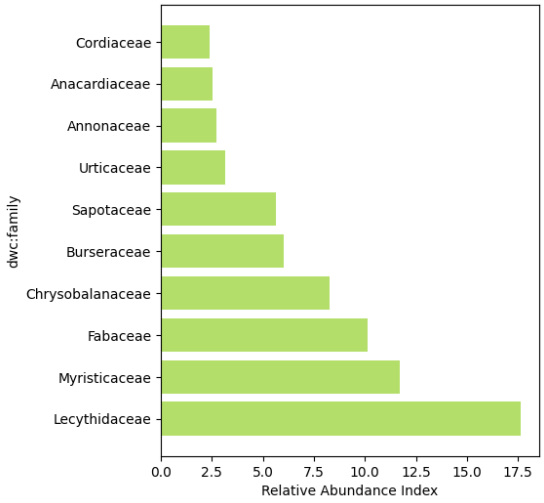
Results. Figure 7 displays a graphical representation of the resulting graph after evaluating the rule r_3 concerning the inference of the Relative Abundance Index. The rule was applied using the REST API detailed in Section 4.3. The service that invokes this rule was called four times one for each BAFOG site. The service is available to explore different spatial objects in the knowledge base, however in this demonstration we report the results for the four given sites. The Y axis expresses the family using the Darwin Core Vocabulary (DWC), whereas the X axis shows the value of the Relative Abundance Index expressed as a percentage relative to the total trees in the given region. Although the regions were spatially close, the subfigures enable us to check that the dominant species/families vary notably between sites. Subfigure 7a displays two dominant families: *Fabaceae* and *Arecaceae*, comprising about 50% of the total families. The other families do not overcome the 7%. Subfigure 7b presents as dominant again the family *Fabaceae* with about 22% of presence. Conversely, the rest of the trees are distributed more uniformly. Subfigure 7c shows major diversity in terms of species, whereas Subfigure 7c concentrates the 40% on only four families: *Myristicaceae*, *Lecythidaceae*, *Fabaceae Burseraceae*. Globally, the four sites contain different families, which can serve as a proxy for biodiversity. Common families in the different sites includes: *Fabaceae*, *Lecythidaceae* et *Myristicaceae*. These rules, accessible through services, enable a standardised exploration of data with the flexibility of the user-defined regions (i.e., geo:Feature). They allow exploration the forest biodiversity, emphasizing particular regions. Different from ad-hoc implementations, our generic rules allow additional explorations and serve for multiple purposes whilst they are easily extendible. This data exploration demonstrates the potential of the different services of OneForestKB.



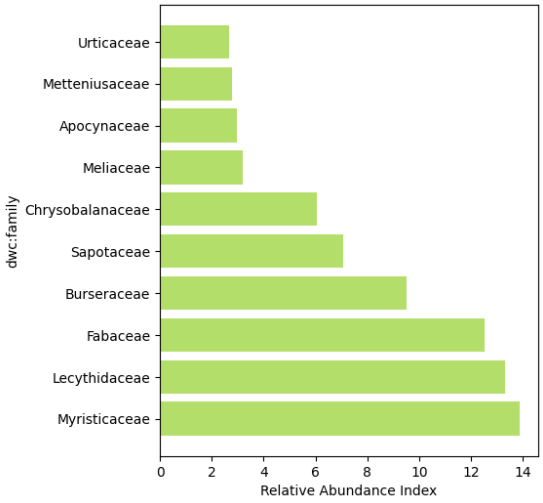
(a) BAFOG_I



(b) BAFOG_II



(c) BAFOG_III



(d) BAFOG_IV

Fig. 7. Relative Abundance Index for the different sites of the BAFOG living lab. For each site, the top ten families of trees are displayed

5.3. Time Performance Analysis

This section studies the time performance of the developed services: validation and enrichment. For this purpose, we define an evaluation protocol to observe processing time. To begin with, we define a new knowledge base exclusively for evaluation containing multiple naming graphs; for each graph, we vary the number of trees. Notice that for both service types, the number of trees is a critical factor affecting processing time; thus, we define seven named graphs for the following numbers of trees: 1000, 2000, 3000, 5000, 7000, 9000, 11000, 13000, 14542. The graph was constructed using Java code, considering that adding a limit to the queries (e.g., LIMIT 1000) does not guarantee a uniform distribution of trees in sites. Secondly, we incorporate to the API a new optional parameter, the context. In this regard, a service call to test the graph with one thousand trees will be as follow:

Listing 5: Curl command to call the sfWithin validation service

```
SERVICE="<endpoint>/validate/sfWithin"
CONTEXT="file%3A%2F%2Ftrees_1000.ttl"
GEO_FEATURE="http%3A%2F%2Fpurl.org%2Feco2adapt%2Flivinglab%2FBAFOG_IV"

curl "${SERVICE}?GeoFeatureURI=${GEO_FEATURE}&context=${CONTEXT}" \
   -H 'accept:␣*/*'
```

For each named graph and for each site in French Guyana (BAFOG_I, BAFOG_II, BAFOG_III, BAFOG_IV) we perform a service call. Besides, we repeat one particular call ten times and calculate the median in order to avoid issues in the latency or the Java garbage collector. We preferred the median over the average considering that the later can be sensitive to the extreme values. To ease reproducibility, the scripts are results are shared in our repository [16]

Validation service sfWithin. Table 5 displays the processing time for the service of validation for the predicate *sfWithin*. In a general view, the table exposes how the number of trees affects the processing time, which seems to be linear up to 14542 trees. Comparing the sites, the time variation is minimal, except for BAFOG_III, which shows an increased processing time due to the fact that this site contains 4098 trees, whereas the other sites have about 3500.

Enrichment services. Table 6 presents the processing time for three different services: Absolute Abundance Index (A_a), Relative Abundance Index (A_r) and Count. These features are implemented as service in Java Spring Boot and as rules in SHACL Advanced Features (SHACL-AF). Count and A_a are independent rules instead A_r requires the two other rules to be calculated. Similar to the validation service, the number of trees is related to the processing time. However, the effect is less representative given that in these services only algebraic computations are evaluated, in contrast to the validation service, from which a geographic function is evaluated. For these services and rules the system demonstrates its efficiency.

[16] https://github.com/felipe-vargas-rojas/OneForestKB

Site # of trees	BAFOG_I	BAFOG_II	BAFOG_III	BAFOG_IV
1000	0.55	0.44	0.84	0.43
3000	1.58	2.01	1.80	1.68
5000	3.08	2.76	3.51	2.73
7000	5.22	4.51	7.23	4.68
9000	9.52	6.75	11.67	7.19
11000	13.12	10.29	15.24	10.82
13000	17.44	15.11	22.14	14.61
14542	21.53	22.98	33.39	18.58

Table 5
sfWithin validation service: processing time for each site in French Guiana varying the number of trees. Time in seconds.

	BAFOG_I			BAFOG_II			BAFOG_III			BAFOG_VI		
	Count	A_a	A_r	Count	A_a	A_r	Count	A_a	A_r	Count	A_a	A_r
# of trees												
1000	0.27	0.33	0.28	0.31	0.31	0.28	0.35	0.37	0.29	0.36	0.26	0.27
3000	0.69	0.62	0.64	0.56	0.58	0.59	0.67	0.68	0.68	0.58	0.58	0.60
5000	1.02	1.02	1.04	0.93	0.95	0.96	1.09	1.09	1.10	0.99	1.00	1.01
7000	1.44	1.45	1.47	1.39	1.40	1.43	1.63	1.61	1.64	1.40	1.41	1.43
9000	1.99	2.00	2.02	1.83	1.84	1.87	2.17	2.19	2.21	1.86	1.88	1.90
11000	2.54	2.57	2.60	2.32	2.36	2.37	2.80	2.82	2.85	2.41	2.42	2.44
13000	3.18	3.18	3.26	2.90	2.92	2.94	3.50	3.52	3.55	2.96	3.00	3.01
14542	3.65	3.68	3.72	3.36	3.38	3.38	4.04	4.07	4.12	3.44	3.46	3.50

Table 6
Processing time of the enrichment services: Absolute Abundance Index (A_a), Relative Abundance Index (A_r) and Count for a given site (e.g., BAFOG_IV). AAI and Count are independent rules, whilst RAI rule needs AAI and Count to be materialised. Time in seconds.

6. Discussion

OneForestKB is presented as a semantic profile along with concrete implementations to tackle diverse use cases concerning forestry experiments. We position a set of ontologies, vocabularies, and design decisions that we support with our use cases, demonstrating how the data can be structured. We maintain the methodology of producing a semantic profile without introducing no new terms, and we continue to uphold that philosophy. The prototype is being used by the partners in the project and is an evolving product. By comparing OneForestKB with similar studies, we can adopt their experiences and lessons learned in a collaborative spirit. Here, we point out two topics of discussion that may contribute to establishing a durable global forest knowledge base.

Scalability Strategy. As shown in Tables 5 and 6, the processing time of both services (geographic validation and enrichments) seems to increase linearly with the number of trees in the knowledge base. For the scalability of OneForestKB, which is intended to be a global knowledge base handling numerous trees, sites, and living labs. We define the scalability plan in three folds: (i) sharding based on living labs, (ii) sharding based on sites, (iii) geographical and predicate indexation.

Sharding is the process of dividing a database into multiple smaller pieces that are easier to manage. In our settings, living labs represent non-overlapping geographic areas. In most use cases, service computation is independent across living labs. Therefore, to address scalability issues that may arise when dealing with multiple living labs, we propose to leverage sharding strategies that divides the knowledge base into smaller, more manageable fragments representing independent knowledge at the living and/or site levels [5]. For instance, if we shard the current living lab of French Guyana and its sites, the processing time for the validation service in Table 5 will not exceed 5 seconds

for the complete graph (containing 14542) since each site has about 3500 trees. To implement sharding [54], in a triple store such as RDF4J, we may leverage the notion of a named graph to store data and the notion of a federated query to target a particular sharding.

Finally, we also plan to add indexation in order to optimize queries. At the moment, we are considering geographic indexes that enable us to compute geographic functions more efficiently, as well as enable more optimal queries involving spatial filters. RDF4J is not a native spatio-temporal triplestore, unlike Parliament [8] or Strabo [10]. Nevertheless, it provides GeoSPARQL-related capabilities through the rdf4j-query-algebra-geosparql module. RDF4J relies on Spatial4J and JTS for geometric processing, and although it currently lacks a dedicated spatial index, future extensions could leverage JTS's STRtree spatial indexing structure. Doing so would significantly improve the performance of SPARQL queries involving topological and spatial-relational functions.

Potential Use Cases. OneForestKB is intended to be extensible, and potential use cases have been considered. One area of ongoing work lies in the necessity to store and present derived data. For example, machine learning can provide approximate results for annotating plant traits. At a different level, there are models that can calculate the percentage species coverage for a particular plot. Similarly, a high-throughput manner of studying forests is to leverage satellite information and a remote sensing approach. Semantic web standards have the potential to provide better tools for this kind of study, as presented in the SorsOnt work [2]. Finally, an interesting use case for different partners is the formalisation and annotation of regions concerning their ecosystem services. A tool with these functionalities could facilitate ecological and climate change studies.

7. Conclusions and Future Work

We have presented the key principles governing both the construction and use of the OneForestKB. We defined a meta-model or semantic profile based on existing data and an initial knowledge base instance in line with the profile. The system is prepared to integrate data from multiple sources, and it is capable of offering services for data quality assurance and data enrichment. We choose widely-adopted referential ontologies from ecology and the OBO foundry ecosystem, and we show how these annotations permit the description of a wide range of aspects about forestry datasets.

Two use cases demonstrate the exploitation of the annotated graph data: one concerns data quality and validation, and the other is intended for data enrichment. The knowledge base was populated with data from two French Guiana living lab sources. We implemented closed-world rules that help to contextualise the particular use cases. For the validation, we constructed rules mixing SHACL and GeoSPARQL enabling us to detect 279 trees that presented erroneous geospatial values. For the enrichment, we propose rules based on SHACL and indicator computations. The rules expressed within SHACL shapes enabled us to visualise the biodiversity in terms of trees' families from four different sites. The rules output in the form of a graph was explored and reported; we noticed the diversity in the different sites and the fact of enabling the user to decide the geographic region where to perform the indicator calculation.

We also build a web interface using React application components, from which we expose some visual examples of the different functionalities, notably the RDF4J server for manipulating the KG data and a web user interface to display plots and visualise maps concerning the regions, livings lab site and trees studied in this work. OneForestKB may be distributed, and several project partners may instantiate its various nodes.

As future work, we plan to incorporate more living labs collections and to explore additional use cases, for instance, derived data and model outputs. In order to study more diverse scenarios and provide more complex analysis, we also plan to integrate open data from sources such as GBIF, public information of eco-regions, and soil maps.

Supplementary Material. Novel SHACL shapes for geospatial data are shared in: https://purl.org/geoshape. The website is available here: https://purl.org/oneforestkb-bafog/app. The REST API can be accessed here: https://purl.org/oneforestkb-bafog/services. Additional materials are added in the repository: https://github.com/felipe-vargas-rojas/OneForestKB

Acknowledgements

This work is supported by the European Horizon Project "eco2adapt" [grant agreement ID 101059498] (eco2adapt: Ecosystem-based Adaptation and Changemaking to Shape, Protect and Maintain the Resilience of Tomorrow's Forests).

References

[1] H. Abelson, B. Adida, M. Linksvayer and N. Yergler, *CC REL: The creative commons rights expression language*, 2012. doi:10.11647/obp.0019.10.

[2] G. Albamonte, G. Falcone, M. Monaco and S. Senatore, Constructing a knowledge base from remote sensing indicators for deforestation assessment **55**(15) (2025), 1014. doi:10.1007/s10489-025-06896-2.

[3] M. Aldwairi, M. Jarrah, N. Mahasneh and B. Al-khateeb, Graph-based data management system for efficient information storage, retrieval and processing **60**(2) (2023), 103165. doi:10.1016/j.ipm.2022.103165. https://linkinghub.elsevier.com/retrieve/pii/S0306457322002667.

[4] E. Arnaud, L. Cooper, R. Shrestha, N. Menda, R.T. Nelson, L. Matteis, M. Skofic, R. Bastow, P. Jaiswal, L. Mueller and G. McLaren, Towards a reference plant trait ontology for modeling knowledge of plant traits and phenotypes, 2012. doi:10.5220/0004138302200225.

[5] A. Azzam, A. Polleres, J.D. Fernández and M. Acosta, smart-KG: Partition-Based Linked Data Fragments for querying knowledge graphs, *Semantic Web* **15**(5) (2024), 1791–1835. doi:10.3233/SW-243571.

[6] T. Baker, Libraries, languages of description, and linked data: A Dublin Core perspective, *Library Hi Tech* **30** (2012). doi:10.1108/07378831211213256.

[7] A.P. Ballantyne, C.B. Alden, J.B. Miller, P.P. Tans and J.W.C. White, Increase in observed net carbon dioxide uptake by land and oceans during the past 50 years **488**(7409) (2012), 70–72. doi:10.1038/nature11299. https://www.nature.com/articles/nature11299.

[8] R. Battle and D. Kolas, Enabling the geospatial Semantic Web with Parliament and GeoSPARQL, *Semant. Web* **3**(4) (2012), 355–370–.

[9] T. Berners-Lee, J. Hendler and O. Lassila, The semantic web, *Scientific american* **284**(5) (2001), 34–43.

[10] D. Bilidas, T. Ioannidis, N. Mamoulis and M. Koubarakis, Strabo 2: Distributed Management of Massive Geospatial RDF Datasets, in: *The Semantic Web – ISWC 2022*, Vol. 13489, U. Sattler, A. Hogan, M. Keet, V. Presutti, J.P.A. Almeida, H. Takeda, P. Monnin, G. Pirrò and C. d'Amato, eds, Springer International Publishing, pp. 411–427, Series Title: Lecture Notes in Computer Science. ISBN 978-3-031-19432-0 978-3-031-19433-7. doi:10.1007/978-3-031-19433-7_24.

[11] F. Bravo Oviedo, C. Ordóñez Alonso and W. Lara Henao, basifoR: paquete de r para manejar los datos del inventario forestal nacional, in: *VIII congreso forestal español*, Sociedad Española de Ciencias Forestales Lleida, 2022.

[12] R. Bruskiewich, E.H. Coe, P. Jaiswal, S. McCouch, M. Polacco, L. Stein, L. Vincent and D. Ware, The plant ontology™ Consortium and plant ontologies, Vol. 3, 2002. ISSN 15316912. doi:10.1002/cfg.154.

[13] P.L. Buttigieg, E. Pafilis, S.E. Lewis, M.P. Schildhauer, R.L. Walls and C.J. Mungall, The environment ontology in 2016: Bridging domains with increased scope, semantic density, and interoperation, *Journal of Biomedical Semantics* **7** (2016). doi:10.1186/s13326-016-0097-6.

[14] R. Chandra, S. Agarwal and N. Singh, Semantic sensor network ontology based decision support system for forest fire management **72** (2022), 101821. doi:10.1016/j.ecoinf.2022.101821. https://linkinghub.elsevier.com/retrieve/pii/S1574954122002710.

[15] J. Corman, J.L. Reutter and O. Savković, Semantics and Validation of Recursive SHACL, in: *The Semantic Web – ISWC 2018*, Vol. 11136, D. Vrandečić, K. Bontcheva, M.C. Suárez-Figueroa, V. Presutti, I. Celino, M. Sabou, L.-A. Kaffee and E. Simperl, eds, Springer International Publishing, Cham, 2018, pp. 318–336, Series Title: Lecture Notes in Computer Science. ISBN 978-3-030-00670-9 978-3-030-00671-6. doi:10.1007/978-3-030-00671-6_19.

[16] J. Corman, F. Florenzano, J.L. Reutter and O. Savković, Validating Shacl Constraints over a Sparql Endpoint, in: *The Semantic Web – ISWC 2019*, Vol. 11778, C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. Cruz, A. Hogan, J. Song, M. Lefrançois and F. Gandon, eds, Springer International Publishing, Cham, 2019, pp. 145–163, Series Title: Lecture Notes in Computer Science. ISBN 978-3-030-30792-9 978-3-030-30793-6. doi:10.1007/978-3-030-30793-6_9.

[17] W. Davis and C.R. Hunt, Knowledge graphs for seismic data and metadata, *Applied Computing and Geosciences* **21** (2024), 100151. doi:https://doi.org/10.1016/j.acags.2023.100151. https://www.sciencedirect.com/science/article/pii/S259019742300040X.

[18] C. Debruyne and K. McGlinn, Reusable SHACL Constraint Components for Validating Geospatial Linked Data (short paper), in: *GeoLD@ESWC*, 2021. https://api.semanticscholar.org/CorpusID:245838256.

[19] A. Dubey, Encyclopedia Britannica: species abundance, 2023, Accessed: 20.03.2025. https://www.britannica.com/science/species-abundance.

[20] ENoLL, Living Labs, 2025, Accessed: 2025-02-26. https://enoll.org/living-labs.

[21] G. Forzieri, L.P. Dutrieux, A. Elia, B. Eckhardt, G. Caudullo, F.A. Taboada, A. Andriolo, F. Bălăcenoiu, A. Bastos, A. Buzatu, F.C. Dorado, L. Dobrovolný, M.-L. Duduman, A. Fernandez-Carrillo, R. Hernández-Clemente, A. Hornero, S. Ionuț, M.J. Lombardero, S. Junttila, P. Lukeš, L. Marianelli, H. Mas, M. Mlčoušek, F. Mugnai, C. Nețoiu, C. Nikolov, N. Olenici, P.-O. Olsson, F. Paoli, M. Paraschiv, Z. Patočka, E. Pérez-Laorga, J.L. Quero, M. Rüetschi, S. Stroheker, D. Nardi, J. Ferenčík, A. Battisti, H. Hartmann, C. Nistor, A. Cescatti and P.S.A. Beck, The Database of European Forest Insect and Disease Disturbances: DEFID2, *Global Change Biology* **29**(21) (2023), 6040–6065. doi:https://doi.org/10.1111/gcb.16912. https://onlinelibrary.wiley.com/doi/abs/10.1111/gcb.16912.

[22] A.S. Foundation, Apache Jena, 2021, Accessed: 2025-02-21. https://jena.apache.org/.

[23] D. Garijo, O. Corcho and M. Poveda-Villalón, FOOPS!: An Ontology Pitfall Scanner for the FAIR Principles **2980** (2021). http://ceur-ws.org/Vol-2980/paper321.pdf.
[24] J.M. Giménez-García, G. Vega-Gorgojo, C. Ordóñez, N. Crespo-Lera and F. Bravo, Improving availability and utilization of forest inventory and land use map data using Linked Open Data **7** (2024), 1329812. doi:10.3389/ffgc.2024.1329812.
[25] G.V. Gkoutos, C. Mungall, S. Dölken, M. Ashburner, S. Lewis, J. Hancock, P. Schofield, S. Köhler and P.N. Robinson, Entity/quality-based logical definitions for the human skeletal phenome using PATO, 2009. doi:10.1109/IEMBS.2009.5333362.
[26] B. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider and U. Sattler, OWL 2: The next step for OWL, *Web Semantics: Science, Services and Agents on the World Wide Web* (2008). doi:http://dx.doi.org/10.1016/j.websem.2008.05.001. http://www.comlab.ox.ac.uk/people/ian.horrocks/Publications/download/2008/CHMP+08.pdf.
[27] N. Hamed, O. Rana, P. Orozco Ter Wengel, B. Goossens and C. Perera, FooDS: Ontology-based Knowledge Graphs for Forest Observatories **3**(1) (2025), 1–42. doi:10.1145/3707637.
[28] J.R. Hobbs and F. Pan, Time Ontology in OWL, Technical Report, W3C working, 2006. http://www.w3.org/TR/owl-time.
[29] R. Hodgson, P.J. Keller, J. Hodges and J. Spivak, QUDT - Quantities, Units, Dimensions and Data Types Ontologies, *W3C* (2014).
[30] A. Hogan, C. Gutierrez, M. Cochez, G.D. Melo, S. Kirrane, A. Polleres, R. Navigli, A.-C.N. Ngomo, S.M. Rashid, L. Schmelzeisen, S. Staab, E. Blomqvist, C. d'Amato, J.E.L. Gayo, S. Neumaier, A. Rula, J. Sequeda and A. Zimmermann, *Knowledge Graphs*, *Synthesis Lectures on Data, Semantics, and Knowledge*, Springer International Publishing. ISBN 978-3-031-00790-3 978-3-031-01918-0. doi:10.1007/978-3-031-01918-0.
[31] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosofand and M. Dean, Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission, W3C, 2004.
[32] J. Hwang, K.W. Nam and K.H. Ryu, Designing and implementing a geologic information system using a spatiotemporal ontology model for a geologic map of Korea, *Computers & Geosciences* **48** (2012), 173–186. doi:https://doi.org/10.1016/j.cageo.2012.05.005. https://www.sciencedirect.com/science/article/pii/S0098300412001616.
[33] R.C. Jackson, J.P. Balhoff, E. Douglass, N.L. Harris, C.J. Mungall and J.A. Overton, ROBOT: A Tool for Automating Ontology Workflows **20**(1) (2019), 407. doi:10.1186/s12859-019-3002-3.
[34] K. Janowicz, A. Haller, S.J.D. Cox, D. Le Phuoc and M. Lefrançois, SOSA: A lightweight ontology for sensors, observations, samples, and actuators **56** (2019-05), 1–10, Publisher: Elsevier BV. doi:10.1016/j.websem.2018.06.003.
[35] G. Klyne and J.J. Carroll, Resource Description Framework (RDF): Concepts and Abstract Syntax, Publisher: W3C Published: W3C Recommendation. http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/.
[36] H. Knublauch and D. Kontokostas, Shapes constraint language (SHACL), Technical Report, W3C, 2017. https://www.w3.org/TR/shacl/.
[37] A.S. Lippolis, G. Lodi and A.G. Nuzzolese, The Water Health Open Knowledge Graph **12**(1) (2025-02-15), 274. doi:10.1038/s41597-025-04537-4. https://www.nature.com/articles/s41597-025-04537-4.
[38] J. Madin, S. Bowers, M. Schildhauer, S. Krivov, D. Pennington and F. Villa, An ontology for describing and synthesizing ecological observation data **2**(3) (2007), 279–296. doi:https://doi.org/10.1016/j.ecoinf.2007.05.004. https://www.sciencedirect.com/science/article/pii/S1574954107000362.
[39] B. Magagna, I. Rosati, M. Stoica, S. Schindler, G. Moncoiffe, A. Devaraju, J. Peterseil and R. Huber, The I-ADOPT Interoperability Framework for FAIRer data descriptions of biodiversity (2021).
[40] W. Nejdl, M. Wolpers, C. Capelle, R. Wissensverarbeitung et al., The RDF schema specification revisited (2000), Publisher: Citeseer.
[41] F.B. Oviedo, J.C.R. González, J.A.M. Núñez and C.O. Alonso, BASIFOR 2.0: Aplicación informática para el manejo de las bases de datos del inventario forestal nacional (2004). doi:10.31167/csef.v0i18.9466. http://secforestales.org/publicaciones/index.php/cuadernos_secf/article/view/9466.
[42] M. Perry, J. Herring, N.J. Car, T. Homburg, S. J.D. Cox, M. Bonduel and F. Knibbe, OGC GeoSPARQL - A geographic query language for RDF data: GeoSPARQL 1.1 draft, Technical Report, OGC implementation standard draft, 2011. https://opengeospatial.github.io/ogc-geosparql/.
[43] E. Prud'hommeaux, S. Harris and A. Seaborne, SPARQL 1.1 Query Language. http://www.w3.org/TR/sparql11-query.
[44] J. Rüegg, C. Gries, B. Bond-Lamberty, G.J. Bowen, B.S. Felzer, N.E. McIntyre, P.A. Soranno, K.L. Vanderbilt and K.C. Weathers, Completing the data life cycle: using information management in macrosystems ecology research **12**(1) (2014), 24–30. doi:10.1890/120375.
[45] O. Software, Virtuoso Universal Server, 2022, Accessed: 2025-02-21. http://virtuoso.openlinksw.com.
[46] F. Tardieu, L. Cabrera-Bosquet, T. Pridmore and M. Bennett, Plant Phenomics, From Sensors to Knowledge, *Current Biology* **27**(15) (2017), R770–R783. doi:https://doi.org/10.1016/j.cub.2017.05.055. https://www.sciencedirect.com/science/article/pii/S0960982217306218.
[47] A. Telenius, Biodiversity information goes public: GBIF at your service **29**(3) (2011), 378–381. doi:10.1111/j.1756-1051.2011.01167.x.
[48] J.T. Tennis, Metadata Application Profiles, in: *Encyclopedia of Archival Science*, L. Duranti and P. Franks, eds, Rowman & Littlefield, 2015, Available at SSRN: https://ssrn.com/abstract=3225431. https://ssrn.com/abstract=3225431.
[49] TERN, TERN Ontology, Technical Report, Terrestrial Ecosystems Research Network (TERN), 2022. https://linkeddata.tern.org.au/information-models/tern-ontology.
[50] M. Uschold and M. Grüninger, Ontologies: principles, methods and applications, *The Knowledge Engineering Review* **11** (1996), 93–136. https://api.semanticscholar.org/CorpusID:2618234.
[51] F. Vargas-Rojas, L. Cabrera-Bosquet and D. Symeonidou, QAVAN: Query-answering approach for actionable numerical relationships over Knowledge Graphs, *Knowledge-Based Systems* **284** (2024), 111252. doi:https://doi.org/10.1016/j.knosys.2023.111252. https://www.sciencedirect.com/science/article/pii/S0950705123010018.
[52] G. Vega-Gorgojo, J.M. Giménez-García, C. Ordóñez and F. Bravo, Pioneering easy-to-use forestry data with Forest Explorer, *Semantic Web* **13**(2) (2022), 147–162.

[53] D. Vrandečić and M. Krötzsch, Wikidata: a free collaborative knowledgebase, *Communications of the ACM* **57**(10) (2014), 78–85.
[54] Z. Wang, G. Sun, N. Wang, L. Gao, C. Xu, Y. Gu, G. Yu and Z. Tian, Lightweight Graph Partitioning Enhanced by Implicit Knowledge, *IEEE Transactions on Computers* **74**(12) (2025), 4153–4167. doi:10.1109/TC.2025.3612730.
[55] J. Wieczorek, D. Bloom, R. Guralnick, S. Blum, M. Döring, R. Giovanni, T. Robertson and D. Vieglais, Darwin core: An evolving community-developed biodiversity data standard, *PLoS ONE* **7** (2012). doi:10.1371/journal.pone.0029715.
[56] M.D. Wilkinson, M. Dumontier, I.J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L.B. da Silva Santos, P.E. Bourne et al., The FAIR guiding principles for scientific data management and stewardship, *Scientific data* **3**(1) (2016), 1–9, Publisher: Nature Publishing Group.
[57] J. Wu, F. Orlandi, D. O'Sullivan and S. Dev, An Ontology Model for Climatic Data Analysis **abs/2106.03085** (2021). https://arxiv.org/abs/2106.03085.
[58] J. Wu, F. Orlandi, D. O'Sullivan and S. Dev, LinkClimate: An interoperable knowledge graph platform for climate data **169** (2022), 105215. doi:10.1016/j.cageo.2022.105215. https://linkinghub.elsevier.com/retrieve/pii/S0098300422001649.
[59] J. Wu, F. Orlandi, D. O'Sullivan and S. Dev, Publishing Climate Data as Linked Data Via Virtual Knowledge Graphs, in: *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 4090–4093. doi:10.1109/IGARSS46834.2022.9884226.
[60] R. Zhu, C. Shimizu, S. Stephen, L. Zhou, L. Cai, G. Mai, K. Janowicz, M. Schildhauer and P. Hitzler, SOSA-SHACL: Shapes Constraint for the Sensor, Observation, Sample, and Actuator Ontology, in: *Proceedings of the 10th International Joint Conference on Knowledge Graphs*, IJCKG '21, Association for Computing Machinery, 2022, pp. 99–107, event-place: Virtual Event, Thailand. ISBN 978-1-4503-9565-6. doi:10.1145/3502223.3502235.
[61] D.L. McGuinness and F. van Harmelen (eds), OWL Web Ontology Language Overview, W3C Recommendation, World Wide Web Consortium, 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/.
[62] B. Motik, P. Patel-Schneider and B. Parsia (eds), OWL 2 Web Ontology Language. Structural Specification and Functional-Style Syntax (Second Edition) (2012). http://www.w3.org/TR/owl2-syntax/.

Appendix A. Full list of prefixes

```
# W3C standards
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix sosa: <http://www.w3.org/ns/sosa/>.
# geo
@prefix geo: <http://www.opengis.net/ont/geosparql#>.
@prefix geof: <http://www.opengis.net/def/function/geosparql/>.
# bio
@prefix gemet: <http://www.eionet.europa.eu/gemet/concept/>.
@prefix dwc: <http://rs.tdwg.org/dwc/terms/>.
# local
@prefix ex: <http://www.example.org/> .
@prefix geoshape: <http://www.purl.org/geoshape#>.
@prefix ecoshape: <http://www.purl.org/ecoshape#>.
@prefix guyafor:<http://purl.org/guyafor#>.
```

Appendix B. Construct SPARQL Query to produce the initial graph of the OneForestKBś services

Listing 6: SPARQL Construct query including some special characters to be parametrised during the user service call. First <%s> is parametrised with the user-defined geo-feature, and the other one geo:%s with the Simple Functions predicates basicstyle

```
CONSTRUCT {
  ?s a geo:Feature.
  ?s ex:selectedTargetNode ?selected.# only added when assigned ?selected property
  ?s ?p ?o.
  ?o ?p2 ?o2.
}
WHERE {
  VALUES (?v) {( <%s> )}
  ?s a geo:Feature.
  {# properties of the given geofeature
    FILTER (?s = ?v)
    BIND(true AS ?selected)
    ?s ?p ?o.
    OPTIONAL {?o ?p2 ?o2}.
  }
  UNION
  {# properties of the contained geofeature such as trees and plots
    ?s geo:%s+ ?v.
    ?s ?p ?o.
    OPTIONAL {?o ?p2 ?o2}.
  }
}
```

Appendix C. Ecological Shapes (EcoShapes)

Listing 7: A SHACL shape containing a rule to compute the total trees of a given area

```
ecoshape:inferTotalTrees a sh:NodeShape;
        sh:targetSubjectsOf ex:selectedTargetNode;
        sh:rule [
                a sh:SPARQLRule ;
                rdfs:label "Infer total trees based " ;
                sh:prefixes ex:, rdf:, geo:, geof:, gemet:, sosa:, guyafor:, dwc
                        :;
                sh:construct """ See Listing 8 """];
        sh:order 2.
```

Listing 8: Construct clause for the SHACL shape ecoshape:inferTotalTrees

```
CONSTRUCT {
   $this ex:hasTotalTrees ?totalTrees.
}
WHERE {
    SELECT ( COUNT (?tree) AS ?totalTrees ) $this
    WHERE {
            $this a geo:Feature.

            ?tree rdf:type dwc:Organism;
                  dct:subject gemet:8664;

            geo:sfWithin+ $this.
    }
    GROUP BY $this
}
```

Listing 9: A SHACL shape containing a rule to compute the Absolute Abundance Index of a given area

```
ecoshape:inferAbsoluteAbundanceIndex a sh:NodeShape;
      sh:targetSubjectsOf ex:selectedTargetNode;
       sh:rule [
            a sh:SPARQLRule ;
            rdfs:label "Infer Absolute Abundance (Aa) Index per Site " ;
            sh:prefixes ex:, rdf:, geo:, geof:, gemet:, sosa:, guyafor:, dwc:;
            sh:construct """ See Listing 10 """];
       sh:order 2.
```

Listing 10: Construct clause for the SHACL shape ecoshape:inferAbsoluteAbundanceIndex

```
CONSTRUCT { $this ex:hasAaIndex ?randomURI.
       ?randomURI dwc:family ?family;
       sosa:simpleResult ?TreeCount.}
WHERE {
    SELECT $this ?family ( COUNT (?tree) AS ?TreeCount ) (IRI(
            CONCAT("http://example.org/Aa/", STRUUID())) AS ?
            randomURI)
    WHERE {
         ?tree rdf:type dwc:Organism;
               dct:subject gemet:8664;
               dwc:family ?family ;
               geo:sfWithin+ $this.}
    GROUP BY $this ?family
    ORDER BY ?family}
```

Listing 11: A SHACL shape containing a rule to compute the Relative Abundance Index of a given area

```
ecoshape:inferRelativeAbundanceIndex a sh:NodeShape;
      sh:targetSubjectsOf ex:selectedTargetNode;
      sh:rule [
            a sh:SPARQLRule ;
            rdfs:label "Infer Relative Abundance (Ar) Index per Site " ;
            sh:prefixes ex:, rdf:, geo:, geof:, gemet:, sosa:, guyafor:, dwc:;
            sh:construct """See Listing 12 """];
       sh:order 3 .
```

Listing 12: Construct clause for the SHACL shape ecoshape:inferRelativeAbundanceIndex

```
CONSTRUCT {$this ex:hasArIndex ?randomURI.
      ?randomURI dwc:family ?family;
      sosa:simpleResult ?Ar.}
WHERE { $this ex:hasAaIndex ?Aa.
      ?Aa dwc:family ?family;
      sosa:simpleResult ?TreeCount.
      $this ex:hasTotalTrees ?totalTrees.
      BIND( (?TreeCount*100/?totalTrees) AS ?Ar)
      BIND(IRI(CONCAT("http://example.org/Ar/", STRUUID())) AS ?
            randomURI)}
```