



D3.2 Knowledge Base Service Architecture Specification v2

WP3 – Advanced Analytics and Knowledge Base

Deliverable Lead: IT Innovation

Dissemination Level: Public

Deliverable due date: 31/12/2017

Actual submission date: 22/12/2017

Version 1.1



Document Control Page	
Title	Knowledge Base Service architecture Specification v2
Creator	Gianluca Correndo (IT Innovation)
Description	This document describes the overall aims and objective of using semantic services in support of the data mining and fusion analytics and it provides an overview of the architecture of semantic services to support these aims.
Publisher	EO4wildlife Consortium
Contributors	Gianluca Correndo (IT Innovation)
Creation date	01/11/2017
Type	Text
Language	en-GB
Rights	copyright "EO4wildlife Consortium"
Audience	<input checked="" type="checkbox"/> Public <input type="checkbox"/> Confidential <input type="checkbox"/> Classified
Status	<input type="checkbox"/> In Progress <input type="checkbox"/> For Review <input type="checkbox"/> For Approval <input checked="" type="checkbox"/> Approved

Disclaimer

This deliverable is subject to final acceptance by the European Commission.

The results of this deliverable reflect only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

Statement for open documents

(c) 2017 EO4wildlife Consortium

The EO4wildlife Consortium (<http://eo4wildlife.eu>) grants third parties the right to use and distribute all or parts of this document, provided that the EO4wildlife project and the document are properly referenced.

Table of Contents

EO4wildlife Project Overview	5
Executive Summary	6
1 Introduction	7
1.1 Purpose.....	7
1.2 Related Documents	7
1.3 Related Literature.....	7
1.3.1 Ecology Data	7
1.3.2 Environmental Data	8
1.3.3 Others	9
1.4 Related Standards.....	10
1.4.1 WPS.....	10
1.4.2 NetCDF	10
1.4.3 SPARQL.....	10
1.4.4 RDF.....	11
2 EO4wildlife Knowledge Base Design	12
2.1 Requirements and Objectives.....	12
2.1.1 Data sources related requirements	12
2.1.2 WPS service related requirements	13
2.1.3 Workflow related requirements	14
2.2 Earth Observation Data Modelling	15
2.3 Tracks Modelling.....	15
2.4 WPS Service and Workflow Modelling	16
2.5 Ontological Upper Model	18
2.6 Comparisons and Integration with Available Standards.....	20
3 Knowledge Base Services Architecture	23
3.1 Architecture Overview	23
3.1.1 Component Diagram.....	23
3.1.2 Semantic Reconciliation Service	24
3.1.3 Raster Metadata Manager.....	25
3.1.4 Tracks Metadata Manager.....	26
3.1.5 WPS Metadata Service.....	26
3.1.6 Workflow Metadata Service	27
3.1.7 Data Retrieval Service	27
3.2 Knowledge Base services implementation	29
3.3 Integration with EO4wildlife platform.....	30
3.3.1 CSW endpoint for catalogue	30
3.3.2 WFS/WCS/WMS access to data	31
3.3.3 Linked Data Platform API	32
4 Conclusion	33
References	34

List of Figures

Figure 1: Turtle encoding of an example of Darwin Core annotation.....	8
Figure 2: Chlorophyll concentration SKOS concept (from http://environment.data.gov.au)	8
Figure 3: NetCDF representation of an sst data set	15
Figure 4: NetCDF representation of an animal track data set.....	16
Figure 5: Representation of a shaded relief WPS service description	17
Figure 6: Representation of WPS services composition.....	17
Figure 7: SST raster time series	19
Figure 8: Leatherback turtle tracking data set.	20
Figure 9: WPS process producing hill-shade map from depth.	20
Figure 10: RDF Turtle representation of the CF name “sea_surface_temperature”	21
Figure 11: RDF Turtle representation of the alignment of the MMI CF URI for sea_surface_temperature... ..	22
Figure 12: WP3 Knowledge Base services architecture and their integration with the EO4wildlife platform	23
Figure 13: Semantic Reconciliation service request.....	25
Figure 14: Semantic Reconciliation service response	25
Figure 15: Response XML document to a WCS <i>DescribeCoverage</i> request	25
Figure 16: How data is transmitted to a processing service execution	28
Figure 17: Semantic alignment of variable.....	29
Figure 18: Spatial + Terminology based query	30
Figure 19: WPS ingestion and semantic annotation	30
Figure 20: CSW GetRecords request via HTTP	31

List of Tables

Table 1: Main KB services' and their responsibilities	24
Table 2: Use cases' specific animal taxonomies	26

EO4wildlife Project Overview

EO4wildlife main objective is to bring large number of multidisciplinary scientists such as biologists, ecologists and ornithologists around the world to collaborate closely together while using European Sentinel Copernicus Earth Observation more heavily and efficiently.

In order to reach such important objective, an open service platform and interoperable toolbox will be designed and developed. It will offer high level services that can be accessed by scientists to perform their respective research. The platform front end will be easy-to-use, access and offer dedicated services that will enable them process their geospatial environmental stimulations using Sentinel Earth Observation data that are intelligently combined with other observation sources.

Specifically, the EO4wildlife platform will enable the integration of Sentinel data, ARGOS archive databases and real time thematic databank portals, including Wildlifetracking.org, Seabirdtracking.org, and other Earth Observation and MetOcean databases; locally or remotely, and simultaneously.

EO4wildlife research specializes in the intelligent management big data, processing, advanced analytics and a Knowledge Base for wildlife migratory behavior and trends forecast. The research will lead to the development of web-enabled open services using OGC standards for sensor observation and measurements and data processing of heterogeneous geospatial observation data and uncertainties.

EO4wildlife will design, implement and validate various scenarios based on real operational use case requirements in the field of wildlife migrations, habitats and behavior. These include:

- Management tools for regulatory authorities to achieve real-time advanced decision-making on the protection of protect seabird species;
- Enhancing scientific knowledge of pelagic fish migrations routes, reproduction and feeding behaviours for better species management;
- Enable researchers better understand the movement behaviour of sea turtle populations; and
- Setting up tools to assist marine protected areas and management.

Abbreviations and Glossary

A common glossary of terms for all EO4wildlife deliverables, as well as a list of abbreviations, can be found in the public document “EO4wildlife Glossary” available at EO4wildlife.eu.

Executive Summary

This document gives an overview of the Knowledge Base service architecture and its integration with the overall EO4wildlife system.

The deliverable describes the overall aims and objective of using semantic services in support of the data mining and fusion analytics and it provides an overview of the architecture of semantic services to support these aims.

The architecture of KB services supports a richer description of the data, services, and workflows managed by the overall EO4wildlife architecture but at any point, the meta-data managed is supporting the existing architecture and it does not overlap in responsibilities and functionalities.

Mandatory inputs to this document are D1.1 “Use Case scenarios v1” [1], D2.1 “System architecture and operational scenarios” [2], D2.3 “External interface for data discovery and processing” [3], and the definition of the data connectors and catalogues whose services are tightly connected with the services described in the present document D3.3 “Big Data connectors and catalogue services” [4]. There is also a reference to the technologies adopted in executing the services in D3.5 “Data Mining and High Level Data Fusion Services v1” [5].

Version 2 of the document builds in great part on its first release (D3.1). Main updates relate to the new sections on the *Ontological Upper Model* and about the *Knowledge Base services implementation*. For the rest of the Deliverable an overall revision was performed on the former content, since most of it remains still valid.

1 Introduction

1.1 Purpose

EO4wildlife is an inherently multidisciplinary project which aims at integrating animal tracking data sets coming from different communities with marine environmental data sets coming from a range of satellites. This is accomplished in the context of a cloud based service platform which provides domain targeted services to process these data sources and run data mining and fusion analytics in support of scientific communities.

1.2 Related Documents

The Knowledge Base services are developed in support of the overall EO4wildlife architecture and it is integrated with it for the deployment of the developed capabilities to the other components of the system. Overall aim of the Knowledge Base services is to enhance the meta-data support provided by OGC standards in order to employ data semantics to provide semantic interoperability at the data access level, providing a homogeneous representation of the entities accessing the data to perform analytics.

Therefore, the present document takes as input the deliverable D2.1 “System architecture and operational scenarios” [1] with which must integrate, and the definition of the data connectors and catalogues whose services are tightly connected with the services described in the present document D3.3 “Big Data connectors and catalogue services” [2].

1.3 Related Literature

This section describes past approaches found in literature about semantically representing or annotating environmental and animal related data for the sake of supporting data interoperability and retrieval. In EO4wildlife in general, and in WP3 in particular, semantic annotation are used to provide a homogeneous access to different, and possibly federated, data sources to provide data mining and fusion analytics services with the intended data to fulfil users’ thematic analytical workflows.

1.3.1 Ecology Data

The Ecological Metadata Language (EML) [6] has been developed with the primary purpose of preserving critical metadata about ecological data sets. It is essentially a generic standard for describing tabular data, in addition to a number of other data formats, which is implemented as a series of XML document types.

Metadata languages have also been developed for describing natural history specimen data, such as Darwin Core. The Darwin Core is actually a body of standards built around RDF and the Dublin Core Metadata Initiative. It includes a glossary of terms identified by URIs whose aim is to facilitate the sharing of information about biological diversity by providing reference definitions, examples, and commentaries. The Darwin Core is primarily based on taxa, their occurrence in nature as documented by observations, specimens, samples, and related information.

```
<http://guid.mvz.org/identifications/23459>
  a dwc:Identification;
  dcterms:identifier "http://guid.mvz.org/identifications/23459";
  dwc:identifiedBy "Richard Sage";
  dwc:dateIdentified "2000"^^xsd:gYear;
  dwciri:toTaxon
<http://lsid.tdwg.org/urn:lsid:catalogueoflife.org:taxon:d79c11aa-29c1-102b-
9a4a-00304854f820:col20120721>.
```

Figure 1: Turtle encoding of an example of Darwin Core annotation

Both the Darwin Core and EML metadata standards primarily focus on describing data structure and high-level contextual information (such as who created a data set and when) but lack support to allow the automated interpretation of the annotated content from an applicative and domain point of view.

The Extensible Observation Ontology (OBOE) [7] aims at providing a formal and generic conceptual framework for describing the semantics of observational data sets (i.e., data sets consisting of observations and measurements). OBOE also prescribes a structured approach for organizing domain specific ontologies through the use of extensions. The basic core structure of the OBOE ontology consists of six concepts:

- **Observation:** an event in which one or more measurements are taken.
- **Measurement:** the measured value of a property for a specific object or phenomenon.
- **Entity:** an object or phenomenon on which measurements are made.
- **Characteristic:** the property being measured.
- **Standard:** units and controlled vocabularies for interpreting measured values.
- **Protocol:** the procedures followed to obtain measurements.

Additional classes and properties can extend from this core set to fit domain specific domains.

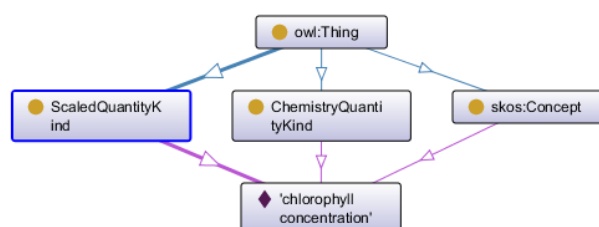
In its basic structure the OBOE ontology achieve similar goals, with a similar approach, to the Observation and Measurement OGC standard [8] and the SSN ontology mentioned in Section 1.3.2.

1.3.2 Environmental Data

[Intro to the section]

Australia's Integrated Marine Observing System (IMOS) is a research infrastructure aimed at supporting the monitoring of Australian oceanic waters [9]. In IMOS system the metadata used to organize the data collections is structured according to the Marine Community Profile (M), a subset and extension of the ISO 19115 standard. The set of vocabularies encoded in the MCP is then provided as linked data via the Australian National Data Service, Research Vocabularies Australia [10].

At the national level, in the pursuit of standardizing the vocabularies used to represent environmental data, and to provide authoritative URIs for vocabularies' entities, the Australian government published a number of RDF vocabularies in linked data [11]. Among these schemes, there are SKOS thesauri [12] to describe observable properties, quantities (e.g. **Chlorophyll concentration** in Figure 2) and unit of measures in order to uniquely identify what is actually contained within data collections.

**Figure 2:** Chlorophyll concentration SKOS concept (from <http://environment.data.gov.au>)

There is some flexibility on how to use those vocabularies to semantically annotate data collections. The most obvious way would be to express the observations themselves as RDF content and use a triple store to extract the needed observations. Due to the size of typical environmental data sets for binary formats

even for limited spatiotemporal extents, this approach is definitely not viable. The translation from binary to RDF format would increase geometrically the size of handled data sets.

There has been some effort to describe earth observations in general, and NetCDF data sets in particular, using ontologies, providing categories for over 300 data collections from climate and other disciplines [12]. These ontologies have been used to describe the NetCDF metadata extracted from data sets in order to support their later retrieval and proved functional to allow mediate data set integration between different systems. The International Research Institute (IRI) for Climate and Society develop a data library (i.e. IRI/LDEO [14]) that employs the aforementioned ontologies to support a climate data portal.

An alternative approach has been proposed by Yu et al. [15] where the metadata section of NetCDF files is enriched by reference URIs from authoritative vocabularies. Using this approach environmental data collections can be annotated with URIs referring to properties, quantities and other contextual information.

Within the European arena, the establishment of authoritative vocabularies to represent environmental data has been the focus of many activities within the INSPIRE initiative and received the support of EC funds, especially in the oceanographic domain [16]. Within the **SeaDataNet** project [17], created a vocabulary server containing SKOS controlled vocabularies covering a broad spectrum of ocean and marine disciplines [18]. The vocabulary server is hosted by the British Oceanographic Data Centre (BODC henceforth), and it provides a linked data access to the vocabularies [19], many of which are developed by the Natural Environment Research Council (NERC henceforth).

A comprehensive classification of terms used in environmental sciences has been provided by the NASA's Jet Propulsion Laboratory with their Semantic Web for Earth and Environmental Terminology (SWEET) ontology [20], [21]. The different modules present in the SWEET ontology describes separate subdomains of Earth and Environmental sciences and provide also a mapping between the properties and quantities encoded in the ontology and a set of NetCDF CF standard names [22].

Also the Marine Metadata Interoperability (MMI) group maintains an extensive repository of ontologies [23]. Among the ontologies served, it is possible to find also the NERC BODC ontologies mentioned earlier.

It is noteworthy that within the Open Geospatial Consortium (OGC), although having an extensive suite of models for describing observations, sensors, environmental data sets, and services, there is only a limited effort in encoding those into ontologies. The only effort in this direction known to the authors is the SSN ontology [24], which integrates modelling patterns from OGC standards into a formal upper ontology and the establishment of extensions to the SPARQL language to include geospatial primitives [25]. Having that said the OGC community is actively exploring how to use more ontologies and controlled vocabularies via a dedicated special interest group [26].

1.3.3 Others

1.3.3.1 Statistical observations

In realms different from the environmental sciences, the World Web Consortium (W3C) has produced a recommendation for an ontology describing multi-dimensional data, the RDF Data Cube vocabulary [27]. The intended use case was describing statistical observations, in fact the RDF Data Cube vocabulary builds in fact upon the core of the SDMX 2.0 model [28], a standard for the encoding and exchange of statistical observations. In the RDF Data Cube vocabulary observations are described as a collections of values characterized with a number of dimensions (e.g. age, region, and year) so that the single observation (e.g. mortality) can be then indexed as at the crossing of each dimension mentioned. Notably Eurostat, provides ontologies and statistical observations encoded in RDF using the aforementioned Data Cube ontology.

This modelling pattern followed by the RDF Data Cube vocabulary is quite similar to the way variables and dimensions in NetCDF files are described and there has been also a proposal to use the Data Cube

vocabulary to describe coverages [29]. As mentioned earlier, adopting this approach within the domain of EO is definitely space consuming since a compact format is then exploded geometrically in all the defining dimensions [19].

1.3.3.2 Provenance

EO4wildlife is not only addressing the semantic annotation of marine observation and animal tracking data sets, but also the services which produces and transforms those data sets in derived artefacts like visual representations of analyses (e.g. graphs or maps), and intermediate results of such analyses. It is important at each step of the data processing workflows to represent where the data come from and how it has been processed to obtain the final results in order to be able to trust the final results.

The W3C has produced a recommendation for an ontology named PROV-O which encodes provenance information [22]. The provenance model encoded in PROV-O can keep track of how entities are generated and modified from initial entities (e.g. data sets) via activities (e.g. by instantiating a WPS process) which involves actors (i.e. single person or an institution) which enact those activities and therefore documenting these transformations and how/who did what [31], [32].

1.4 Related Standards

This section describes the relevant standards involved in the definition of the architecture and its integration with the overall EO4wildlife architecture.

1.4.1 WPS

As described in EO4wildlife deliverable D2.3 [3] the OGC standard for Web Processing Service (WPS) provides rules for the execution of a geospatial process and standardizes the inputs and outputs for geospatial processing services. Alongside the standardization of procedures there are also a number of XSD schemas which allows to provide a service description upon request.

A WPS process description include also, among other documentation, the type of the input/output in terms of mime type.

1.4.2 NetCDF

The Network Common Data Format (NetCDF henceforth) is a set of software libraries and an OGC proposal of self-describing, machine-independent data format that support the creation, access, and sharing of array-oriented scientific data [32]. An in-depth description of the standard has been provided in [4].

1.4.3 SPARQL

SPARQL 1.1 is a set of specifications developed by the W3C consortium that provide languages and protocols to query and manipulate RDF graph content on the Web or in an RDF store. The standard includes the following specifications:

- SPARQL 1.1 RDF Query Language.
- Query results formats: XML, JSON, CSV, and TSV.
- SPARQL 1.1 Federated Query Extension.
- Entailment regimes defining the semantics of SPARQL queries under RDF Schema, OWL, or RIF.
- SPARQL 1.1 Update Language.
- SPARQL 1.1 Full Protocol for RDF defining means for conveying arbitrary SPARQL queries and update requests to a SPARQL service.

- SPARQL 1.1 Service Description defining a method for discovering and a vocabulary for describing SPARQL services.
- SPARQL 1.1 Graph Store HTTP Protocol, as opposed to the full SPARQL protocol, defines minimal means for managing RDF graph content directly via common HTTP operations.
- SPARQL 1.1 Test Cases.

1.4.4 RDF

The Resource Description Framework (RDF) [34] is a data model for representing meta-information, initially designed for expressing metadata for resources in the World Wide Web, it has gradually grown to incorporate many features of a knowledge description language, especially with the combined use of RDF-family languages such as RDFS and OWL.

The mechanism to identify such resources is by associating them with a unique URI and the way to express statements about resources is by asserting statements about these resources in form of triples: subject, predicate, and object. This approach allows to define graphs describing a number of resources which are bind by predicates. The graphs themselves can then be serialized in a particular data format: Turtle, N-Triples, N-Quads, JSON-LD, N3, RDF/XML, and binary also using a format named HDT [35].

2 EO4wildlife Knowledge Base Design

2.1 Requirements and Objectives

This section describes a set of requirements for a common semantic data model and the functionalities that are built on top of these models. The requirements have been organized in three main areas: data sources related requirements, service related requirements, and workflow related requirements. Functional requirements for the KB include forms of reasoning and services to provide. These consequently provide consequently further requirements at the semantic modeling level.

Objective of the Knowledge Base and of the semantic modeling is twofold:

- a) To provide an abstract data access layer which allows access to heterogeneous data sources using common interfaces.
- b) To support a catalogue service which maintains the collection of descriptive metadata that can be searched by users.

The requirements described in this Section are briefly identified with an id to ease the task of referring to them throughout other deliverables. The id is formed by:

EO4-SEM-{incremental number}

EO4wildlife project

..... **SEM**antic services

2.1.1 Data sources related requirements

This section shall describe all the requirements and objectives related to the management of data sources' metadata and the required semantic annotations to provide within the platform. The entities modeled here are the data sources to be processed by the data mining and fusion services and include EO data products (e.g. raster time series of a marine observation) and animal tracks (coming from different providers and normalized during the data ingestion phase).

Main requirement common to all types of data sources handled in EO4wildlife is to model, annotate, and reason over data sources' meta-data only, avoiding to representing the data observations as well as in many approaches described in Section 1.3. Due to the high volumes of data handled, the semantic models shall support the data retrieval using a common upper model, but leaving the implementation of the access to the data itself to other services in the platform.

The set of functional requirements at the data level can be summarized as follows:

EO4-SEM-1

The ontology shall be able to represent datasets, variables, and dimensions including the functional dependencies between dimensions and variables.

EO4-SEM-2

The ontology shall be able to represent annotations of dimensions and variables with reference URIs from a target EO4wildlife controlled vocabulary. The vocabulary shall encode marine observations in the case of EO data sets and animal taxa in the case of tracking data sets.

EO4-SEM-3

The ontology shall be able to represent dependencies between raw observations (e.g. ocean color) and processed observations (e.g. chlorophyll concentration). For EO data sets this requires to introduce a distinction between L1 and L2 data products. For the animal tracking data sets, this requires to represent original position values with their quality levels as they are produced by ARGOS and the processed positions once the data has been processed (e.g. via the Track&Lock algorithm [36])

EO4-SEM-4

The ontology shall be able to represent provenance information: who produced the data set (i.e. person or organization), quality, and processes applied to the data. This information is expected to be represented as metadata using CF conventions.

2.1.2 WPS service related requirements

This section shall describe all the requirements and objectives related to the description management of WPS services' metadata. WPS services here are intended as potential data processing capabilities (i.e. potential functions) which can then be instantiated with ground input data sources and parameters to produce concrete output results.

The set of functional requirements at the WPS service level can be summarized as follows:

EO4-SEM-5

The ontology shall be able to represent a WPS service, its intended input and output parameters' types by making reference towards a reference thesauri of domain entities. This is true for both type of data sources; EO and animal tracks. If a track analysis is meaningful only for a given taxa this should be represented and used when a consistency check is requested. Similarly, for EO data sets, if a procedure makes sense only if applied to a type of data sets (e.g. shading reliefs require a digital elevation model, or DEM) this type constraints shall be explicitly represented in the ontology.

EO4-SEM-6

The ontology shall be able to represent both a WPS service and its instantiation with ground input data sets and parameters once invoked within a workflow. A service and its instantiations are separate entities although the link between the two types of entities must be kept.

EO4-SEM-7

The ontology shall be able to represent quality information associated to the execution of the WPS service so to keep track of the quality information related to its execution on ground data (i.e. model's precision, data granularity, statistical indicators).

EO4-SEM-8

The KB services shall be able to produce the intended data types for the input and output parameters of a WPS service.

EO4-SEM-9

The KB services shall be able to check the consistency of an instantiation of a WPS service with ground input and output data sets and parameters. For example the KB service shall produce warnings when a shaded relief algorithm is run with a SST dataset as input instead of a DEM.

EO4-SEM-10

The KB services shall produce, given a WPS service description and the set of EO data sets currently described in the KB, the possible data sets that are consistent with the intended use of the WPS service. This service shall overcome the present limitation of WPS standard of representing input data types only with mime-types.

EO4-SEM-11

The KB services shall be able to compute the set of constraints implied by the instantiation of a WPS service with ground input and output data sets and parameters. For example the KB service shall compute, by using the data sets and WPS semantic description, the intended bounding box and temporal period when a tracking data is defined as input of a WPS service.

2.1.3 Workflow related requirements

This section shall describe all the requirements and objectives related to the management of workflows' metadata. Workflows here are intended as a chain of potential data processing capabilities that can then be instantiated with concrete input data sources and parameters. As such all the requirements and functionalities devised for WPS services alone can be restated here for WPS services workflows seen as collections.

EO4-SEM-12

The ontology shall be able to represent a WPS workflow, the WPS services which compose the workflow and the dependencies between input and output parameters in the workflow.

EO4-SEM-13

The KB services shall be able to check if an instantiation of a workflow with given input data sets produces some consistency warning.

EO4-SEM-14

The KB services shall be able to check if an instantiation of a workflow with given input data sets produces some warning related to correlation between data sets (e.g. using two data sets in a service where one is derived by the other) .

EO4-SEM-15

The KB services shall be able to compute the set of constraints implied by the instantiation of a workflow with ground input and output data sets and parameters. For example the KB service shall compute, by using the data sets and semantic description of the WPS services which compose a workflow, the intended bounding box and temporal period when a tracking data is defined as input of a WPS service and provide constraints for those.

2.2 Earth Observation Data Modelling

Earth Observation metadata modelling (product level and observation level), integration with domain relevant thesauri and existing OGC standards (extension of OGC standard if and when possible)

The common upper model describing data sources shall abstracts the different types of data by adopting the same conceptualisation which, in this case, is inspired from NetCDF data model. In NetCDF files, the data sets are represented as a set of variables which are functionally dependant from a number of dimensions. The number and nature of the dimensions dictate the type of data set represented.

A raster file (e.g. a bathymetry) which has only one variable whose values depend on the latitude and longitude shall have in NetCDF one variable (i.e. the value of depth measured at that point in the raster grid with an associated unit of measure) which is dependent from two inputs (called “dimensions” in NetCDF), the latitude and the longitude, both associated with a unit of measure and with a given range and granularity. We therefore can represent the bathymetry as a function:

$$\text{bathymetry: } \text{Latitude} \times \text{Longitude} \rightarrow \text{Depth}$$

As a function, the bathymetry will have a given range (i.e. the number of coordinates which fall within the raster grid) and a domain (i.e. in this case the set of depth values collected).

A raster temporal series (e.g. the values of sea surface temperature collected for a time period) has one variable which is dependent from three dimensions: the latitude, the longitude, and the time stamp when the observation was collected (see Figure 3).

We therefore can represent the raster temporal series of sea surface temperature (or **sst**) as a function:

$$\text{sst: } \text{Latitude} \times \text{Longitude} \times \text{Time} \rightarrow \text{Temperature}$$

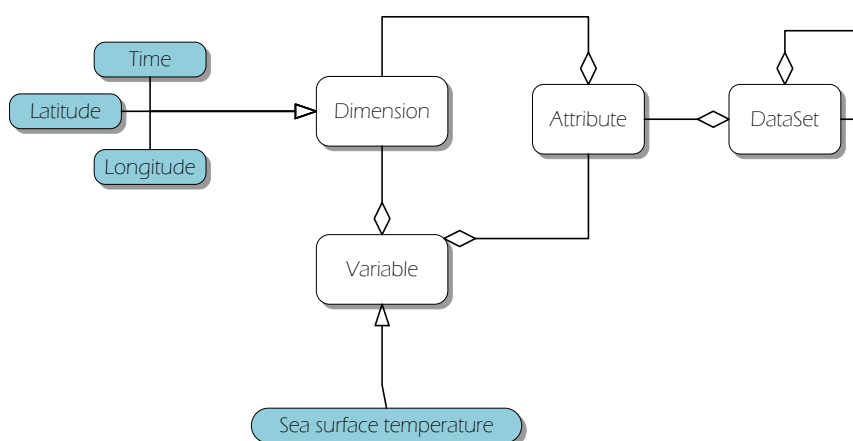


Figure 3: NetCDF representation of an sst data set

2.3 Tracks Modelling

Animal tracks can also be represented as functions in NetCDF (see Figure 4). In particular, an animal track is a data set containing one variable, the position of an animal encoded in some coordinate reference system, depending by two dimensions, namely the id of the animal (assuming the dataset contains many tracks) and the time stamp.

We therefore can represent an animal track data set as a function:

$$track: AnimalId \times Time \rightarrow Position \equiv Latitude \times Longitude$$

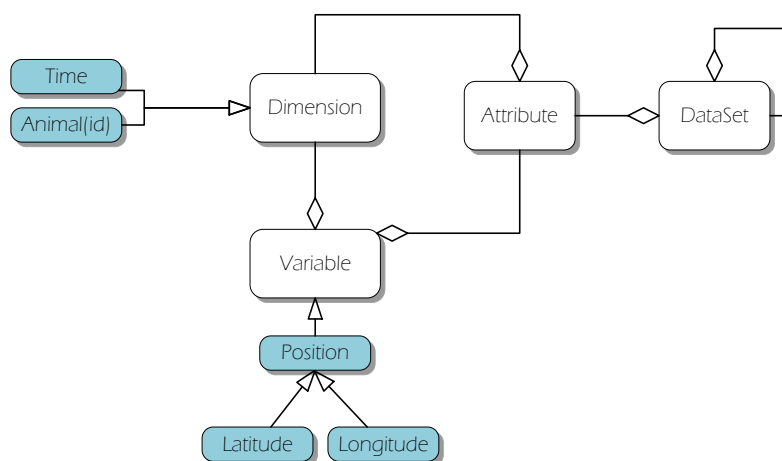


Figure 4: NetCDF representation of an animal track data set

At this level of abstraction the data sets are described in terms of functional dependencies between dimensions and variables. Dimensions and variables constitute the domain and range of the function itself and they are represented as algebraic sets of values (e.g. the set of latitudes in a particular data set) which are annotated with type values aligned with a managed type system (e.g. a reference thesaurus) and unit of measures.

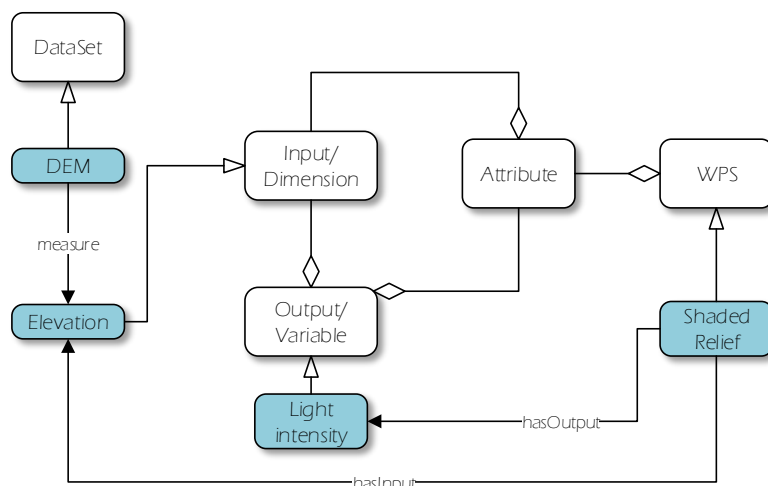
Additionally to the aforementioned annotations (e.g. type of observations, units of measure, extent) it is mandatory to represent a set of attributes identified and used in the data ingestion process, and described in deliverable D3.3 [4]. These attributes shall follow the same CF naming conventions mentioned in D3.3 to avoid complex mappings and ingenerate confusion.

2.4 WPS Service and Workflow Modelling

At this level the services are described in terms of their I/O providing semantic annotations for them in order to provide type checks and formal consistency when the services are instantiated with ground data, and finally to keep track of the functional dependencies and provenance of the results provided. The similarities between WPS services and data sets abstraction based on I/O functional dependencies is intended and it shall allow for a common conceptualization of both which will be seen as specialization of common concepts.

The basic difference between data sets and services is that a data set are usually associated to a bounded ground range and domain (i.e. a precise set of latitudes, longitudes and time stamps for which the data set can provide observations) whereas a service needs to be instantiated with ground inputs to be able to provide outputs.

As per the data sets, it is important to keep track of the services applied to input datasets and how these have produced output results. The ontology describing the WPS services shall include a profile describing the type of algorithms applied.



WPS standard represents workflows as function compositions where the inputs of a service is provided by the execution of some other service. This leads to limited topologies of workflows and allows to represent a workflow as a function composition which in turns allow to reason upon the actual domain and range of the workflow itself. An example of how the composition of WPS services is rendered in this model is depicted in Figure 6 where a species tracks are first filtered before extracting the population distribution via the gridding services.

The functional representation of this composition can be represented as functional composition of two separate functions:

$$track: AnimalId \times Time \rightarrow Position \equiv Latitude \times Longitude$$

$$filter: Latitude \times Longitude \times Quality \rightarrow Position \equiv Latitude \times Longitude$$

$$gridding: Latitude \times Longitude \times Grid \rightarrow Distribution \equiv Latitude \times Longitude \rightarrow AnimalCount$$

$$workflow: track \circ filter \circ gridding$$

2.5 Ontological Upper Model

The above described model is then here modelled using DL axioms that form the building block of the upper ontological models used in the KB services.

The general functional dependency ontology include general concepts for functions, sets, subsets, Cartesian product and the relation between function domain and codomain. Sets are represented here only intentionally, as representative of all potential elements belonging to them, and only in special cases extensionally (i.e. enumerating all their instances), for example when we want to represent sampling grids. This decision is due to the fact that the aim of the semantic representations is to express functional dependencies and to have the means to semantically annotate physical properties such as sea surface temperature with concepts from a domain vocabulary:

```
Set ⊆ T
SubSet ≡ ∃ subsetOf Set
SubSet ⊆ Set
CartesianSet ⊆ Set
Function ⊆ T
  ∃ hasDomain Thing ⊆ Function
  T ⊆ ∀ hasDomain Set
  ∃ hasCodomain Thing ⊆ Function
  T ⊆ ∀ hasCodomain Set
```

EO data sets such as raster time series specialise these general concepts by defining *Dataset* as a subclass of *Function* and *hasDimension* and *hasCodomain* as sub-properties respectively of *hasDomain* and *hasCodomain*:

```
Dataset ⊆ Function
hasDimension ⊆ hasDomain
hasVariable ⊆ hasCodomain
Dataset ⊆ ≥ 1 hasVariable Set
Dataset ⊆ ≥ 1 hasDimension Set
```

Moreover, in order to represents spatial and temporal dimensions homogeneously with the chosen set-based interpretation of functions, *Position*, *Time*, and *TimeInstant* concepts are introduced (note that *Positions* can include for the marine animals who can dive also a depth dimension):

```
Position ⊆ = 1 hasPartSet Longitude
Position ⊆ = 1 hasPartSet Latitude
Position ⊆ CartesianSet
Time ⊆ Set
TimeInstant ⊆ = 1 hasElement Time
TimeInstant ⊆ Time
```

The introduction of spatio-temporal concepts allows us to define *TrackingDataset* concepts as a specialization of *Dataset* (and of *Function* consequently) which relates *Time* instances with *Position* instances, and *RasterTimeSeries* as a specialization of *Dataset* which relates *Position* instances to elements of other sets:

```
TrackingDataset ≡ Dataset ⊓ ∃ hasDimension Time
    ⊓ ∃ hasVariable Position
RasterTimeSeries ≡ Dataset
    ⊓ ∃ hasDimension Latitude
    ⊓ ∃ hasDimension Longitude ⊓ ∃ hasDimension Time
```

Moreover, with the definition of singleton concept *TimeInstant* we can define the *Raster* concept which is a particular type of *RasterTimeSeries* for which we have only one time instant (i.e. the time in which the data has been collected). An example of instance of *Raster* concept is the bathymetry data sets for which the value of depth of the sea bed doesn't change over time:

```
Raster ≡ RasterTimeSeries ⊓ ∃ hasDimension TimeInstant
```

WPS processes are modelled simply as a direct specialisation of the *Function* concept and the *hasDomain* and *hasCodomain* properties:

```
Process ⊆ Function
hasInput ⊆ hasDomain
hasOutput ⊆ hasCodomain
```

Examples of data sets encoded with the presented ontology can be seen depicted in Figure 7, a raster time series providing values of sea surface temperature (SST), in Figure 8, a tracking data set from the sea turtles data base, and in Figure 9, a WPS process producing a hill shade map from a raster bathymetry data measuring depth of sea bed.

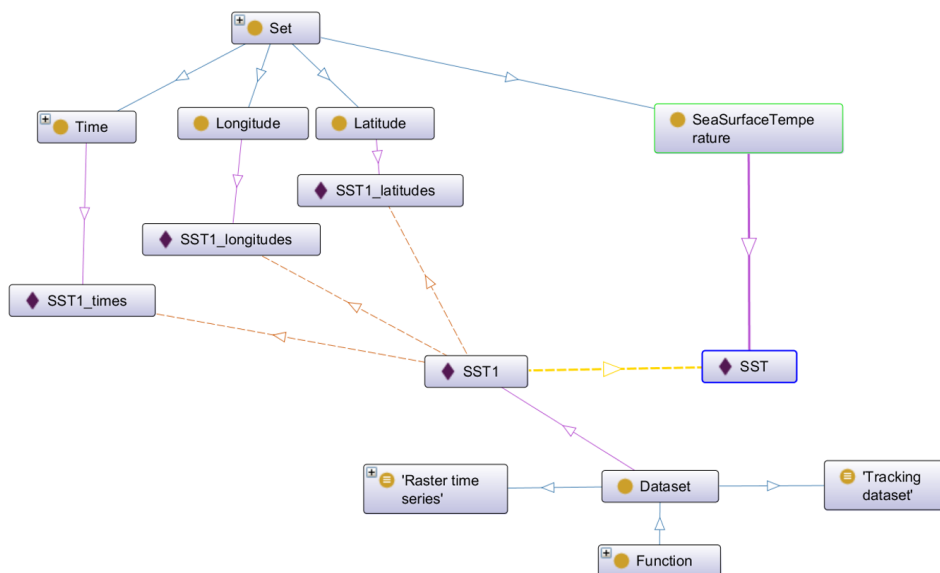


Figure 7: SST raster time series

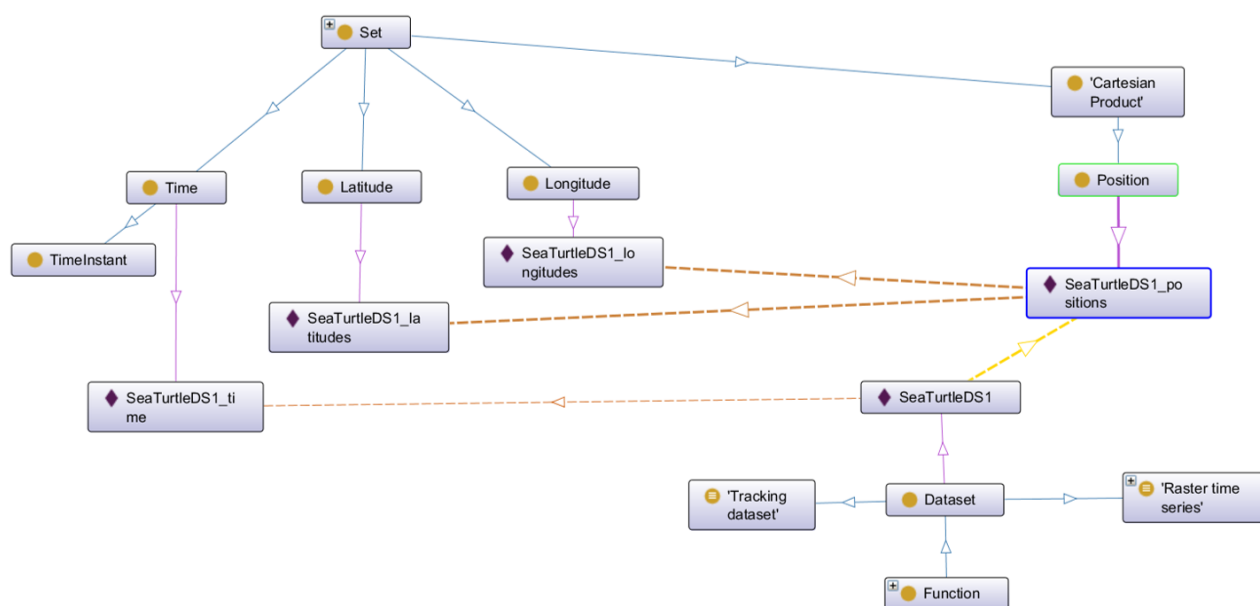


Figure 8: Leatherback turtle tracking data set.

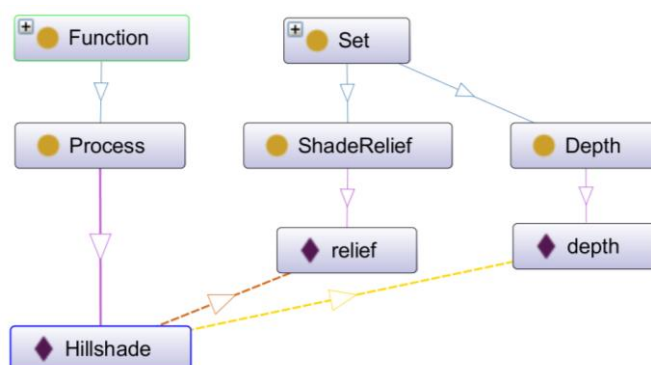


Figure 9: WPS process producing hill-shade map from depth.

Once the domains and codomains have been identified and the functional dependencies between them expressed, a set of information to annotate them has been identified. Annotating the data sets is important because scientists need to know further information about the tracked animal in order to correctly analyse their movements and behaviour. Information such as species, age, breeding and migratory status, and colony location are essential for interpreting the tracking information and for deciding what specific aspect of their behaviour to model. All the above information have been encoded

2.6 Comparisons and Integration with Available Standards

The main standards relevant here are encoded in the CF conventions encoded in D3.3 which are used during the data ingestion phase to represent the metadata within NetCDF files in a more standardized way. The Climate and Forecast (CF) convention for metadata are designed to promote the processing and sharing of NetCDF files by identifying a set of standard names for observations and their measurement units. The conventions define metadata that provide a definitive description of what the data in each

variable represents, and the spatial and temporal properties of the data. This improve the semantic interoperability of data sets when exchanged.

The thesauri encoding the CF conventions can be used here to annotate the observed properties, providing reference URIs for observation types and data products. The Marine Metadata Interoperability Ontology Registry and Repository provides also linked data representations of such conventional names to align to (see Figure 10).

```
@prefix skos:      <http://www.w3.org/2004/02/skos/core#>.
@prefix :         <http://mmisw.org/ont/cf/parameter/>.

:sea_surface_temperature a :Standard_Name ;
    :canonical_units "K" ;
    skos:definition "Sea surface temperature is usually abbreviated as
\"SST\". It is the temperature of sea water near the surface (including the part
under sea-ice, if any). More specific terms, namely
sea_surface_skin_temperature, sea_surface_subskin_temperature, and
surface_temperature are available for the skin, subskin, and interface
temperature. Respectively. For the temperature of sea water at a particular
depth or layer, a data variable of sea_water_temperature with a vertical
coordinate axis should be used.'".
```

Figure 10: RDF Turtle representation of the CF name “sea_surface_temperature”

For the animal tracking data, in D3.3 [2] a set of conventional metadata annotations are provided to represent needed domain meta-data annotation to tracking data in different domains. Due to the fact that different use cases deal with different marine species: birds, fishes, turtles, marine mammals; it is important to represent moving points in three dimensions so that, if made available, it is possible to represent animal diving or height of flight. Moreover, the annotations of animal tracks may include the quality of positions provided by ARGOS satellite and a number of annotations identifying the animal and possibly its species.

The set of conventions adopted within NetCDF files are described in deliverable D3.3 [2] and D3.4 [37].

The EO4wildlife project does not have any requirements on external data providers. These providers are currently compliant with the CF convention but are assigning the standard name variable according to the standard name table (see [38]). The integration of data into the EO4wildlife platform relies on the usage of the knowledge base by converting the CF standard name value into Links Data objects references, Uniform Resource Identifiers (URI). Such references precise the semantics to be used not only for the extraction but also for the display of the variables and their associated attributes.

For instance, the sea surface temperature can be represented as:

- ‘analysed_sst’ in the extraction request to the external provider,
- Downloaded in the NetCDF response as a ‘sea_surface_temperature’ CF variable,
- Identified by the ‘http://mmisw.org/ont/cf/parameter/sea_surface_temperature’ URN,
- The label to display something like: ‘Sea Surface Temperature’, with the following comment: ‘Sea surface temperature is usually abbreviated as "SST". It is the temperature of sea water near the surface (including the part under sea-ice, if any), and not the skin temperature, whose standard name is **surface_temperature**. For the temperature of sea water at a particular depth or layer, a data variable of **sea_water_temperature** with a vertical coordinate axis should be used’.

The EO4wildlife platform will be able to use GeoServer WCS/WFS endpoints to manage tracks data sources, once they will be made available in NetCDF format.

Available sources of linked data URIs describing nomenclatures of relevance to the EO4wildlife project can be identified by the above mentioned Marine Metadata Interoperability (MMI henceforth) Ontology Registry and Repository which provide CF convention terms' URIs but no link to broader/narrower terms aligned from other nomenclatures. Such information can be retrieved by using the alignments from the MMI to the NERC linked data vocabulary (see Figure 11).

```
@prefix skos:      <http://www.w3.org/2004/02/skos/core#>.
@prefix :         <http://mmisw.org/ont/cf/parameter/>.

:sea_surface_temperature a :Standard_Name ;
    skos:exactMatch <http://vocab.nerc.ac.uk/collection/P07/current/CFSN0381>.

<http://vocab.nerc.ac.uk/collection/P07/current/CFSN0381> skos:broader
<http://vocab.nerc.ac.uk/collection/P02/current/TEMP/> ,
<http://vocab.nerc.ac.uk/collection/P04/current/G963/>.
```

Figure 11: RDF Turtle representation of the alignment of the MMI CF URI for sea_surface_temperature

3 Knowledge Base Services Architecture

3.1 Architecture Overview

The architecture is composed of different components which implement the KB services functionalities which are exposed to the remaining platform as REST services. The KB services shall be deployed as a separate Docker container which will declare all needed dependencies. The persistence layer shall be implemented by the Virtuoso triple store instance provided by the SEEED platform.

3.1.1 Component Diagram

This section shall describe a component breakdown of the KB service, identifying the responsibilities of each component. Figure 12 depicts the overall components and their interactions with the services provided by the SEEED architecture and deployed in WP2 (see deliverable D2.1 ‘System architecture and operational scenarios’ for an overview of the SEEED architecture [34]).

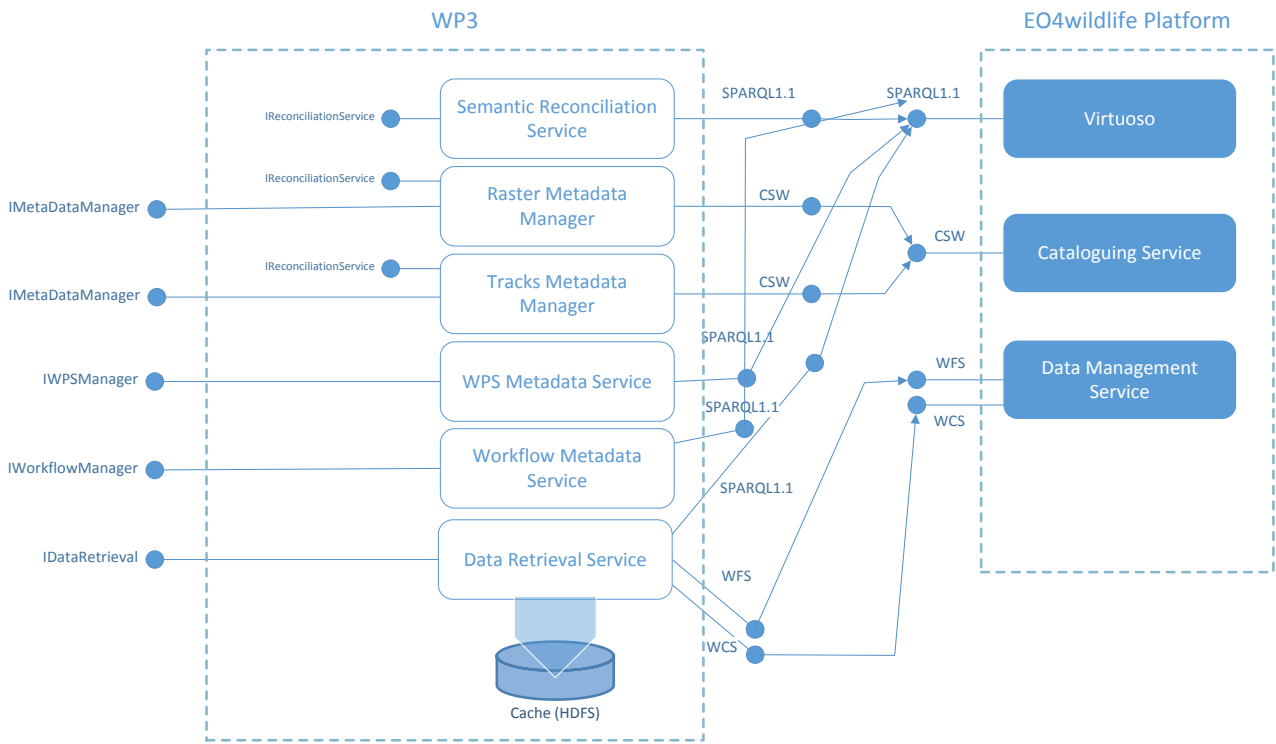


Figure 12: WP3 Knowledge Base services architecture and their integration with the EO4wildlife platform

The following table represents the main components of the KB services:

Service	Responsibilities
Semantic Reconciliation Service	This service reconciles terms (strings) against entities (URI) known by the platform and stored in the platform triple store (the Virtuoso server in Figure 6).
Raster Metadata Manager	This service provides utility functions to inspect NetCDF files and extract meta-information regarding the dimensions, variables, their definition, the functional dependencies, and all relevant annotations needed to align the raster data set to the target ontologies. Additionally, this service implements part of the data related services outlined in Section 2.1.1.
Tracks Metadata Manager	This service provides utility functions to inspect NetCDF files and extract meta-information regarding the dimensions, variables, their definition, the functional dependencies, and all relevant annotations needed to align the animal tracks data set to the target ontologies. Additionally, this service implements part of the data related services outlined in Section 2.1.1.
WPS Metadata Service	This service provides a functional interface to semantically annotate the WPS services managed by the platform and to implement the WPS services' related functionalities outlined in Section 2.1.2.
Workflow Metadata Service	This service provides a functional interface to semantically annotate the WPS services managed by the platform and to implement the WPS services' related functionalities outlined in Section 2.1.3.
Data Retrieval Service	This service implements part of the data related services outlined in Section 2.1.1 relevant to the retrieval of data sets by using semantic queries.

Table 1: Main KB services' and their responsibilities

3.1.2 Semantic Reconciliation Service

This section shall describe the services to semantically reconcile the entities produced from the data ingestion and service and workflow annotation services towards target ontologies and thesauri.

The process of semantic reconciliation is aimed at reconciling a non-semantic piece of information (i.e. a label from a data set description) with a reference ontology or controlled vocabulary. Direct result of the semantic reconciliation is a URI which uniquely defines the input entity within the KB and a standard label for the user to annotate the entity with. Semantic reconciliation is not an automatic process and it must be seen in perspective of supporting users' activities. In particular, the semantic reconciliation services here described are in support of the metadata editing use cases for data, services, and workflows.

The Semantic Reconciliation Service will be implemented as a REST service whose protocol and interface will adhere to the OpenRefine Reconciliation Service API [39] [40]. The service will translate the semantic reconciliation query into a SPARQL query for types and their description to return to the caller.

The service allow for two types of requests. The first request returns a descriptor of the service itself in JSON format, and the second type of request is for reconciling named entities. The reconciliation request is encoded in a JSON object (see Figure 13 for an example) detailing the parameters of the request and passed into the body of the GET request.

```
{
  "q0" : {
    {
      "query" : "Sea surface temperature",
      "limit" : 3,
      "type" : "http://mmisw.org/ont/cf/parameter/StandardName",
      "type_strict" : "any",
      "properties" : [
        { "http://mmisw.org/ont/cf/parameter/canonical_units" : "K"}
      ]
    }
    ...
  }
}
```

Figure 13: Semantic Reconciliation service request

The service can serve more name queries per request, each response is provided in a JSON dictionary object with the same key of the request (see Figure 14).

```
{
  "q0" : {
    "result" : [
      {
        "id" : "http://mmisw.org/ont/cf/parameter/sea_surface_temperature",
        "name" : "sea_surface_temperature",
        "type" : "http://mmisw.org/ont/cf/parameter/StandardName",
        "score" : 0.9
        "match" : true #if the service is quite confident about the match
      }
    ]
    ...
  }
}
```

Figure 14: Semantic Reconciliation service response

3.1.3 Raster Metadata Manager

This section shall describe the set of services that support the extraction of metadata from ingested raster data sets (e.g. NetCDF and CF compliant tags) and assist in the translation of their metadata.

```
<gmlcov:rangeType>
  <swe:DataRecord>
    <swe:field name="analysed_sst">
      <swe:Quantity definition="http://opengis.net/def/property/OGC/0/SST">
        <swe:description>Sea surface temperature</swe:description>
        <swe:uom code="K"/>
      </swe:Quantity>
    </swe:field>
    ...
  </swe:DataRecord>
</gmlcov:rangeType>
```

Figure 15: Response XML document to a WCS *DescribeCoverage* request

The Raster Metadata Manager provides services to extract the meta-data elements from NetCDF files and WCS XML coverage XML documents (see Figure 15) containing raster data (i.e. marine observations),

semantically reconcile those with managed entities' URIs using the Semantic Reconciliation Service described in Section 3.1.2, and synchronize the overall RDF profile with the underlying triple store (see the "Virtuoso" component in Figure 12) using the SPARQL1.1 update protocol.

Moreover, the Raster Metadata Manager shall provide specialized semantic reconciliation services for marine observation data sets to integrate with UI components providing keyword completion services or semantic annotation services to part of the data sets metadata.

3.1.4 Tracks Metadata Manager

This section shall describe the set of services that support the extraction of metadata from ingested animal tracks data sets (stored in NetCDF files as CF compliant tags).

The Tracks Metadata Manager provides services to extract the meta-data elements from NetCDF files containing animal tracks, semantically reconcile those with managed entities' URIs using the Semantic Reconciliation Service described in Section 3.1.2, and synchronize the overall RDF profile with the underlying triple store (see the "Virtuoso" component in Figure 12) using the SPARQL1.1 update protocol.

Moreover, the Raster Metadata Manager shall provide specialized semantic reconciliation services for marine observation data sets to integrate with UI components providing keyword completion services or semantic annotation services to part of the data sets metadata. In particular this service shall provide semantic reconciliation with animal taxonomies for each use case established in D1.1 "Use Case scenarios v1" [1].

Use case	Animal taxonomy
Sea birds	BirdLife provided a taxonomy of the birds of the world [41] which includes classification information such as: order, family, common and scientific name.
Sea turtles	University of Exeter pointed out that there exist a world wide database of reptile taxonomical classifications maintained by volunteers [42].
Pelagic fishes	<i>Still to find a viable taxonomy for pelagic fishes.</i>

Table 2: Use cases' specific animal taxonomies

The Tracks Metadata Manager shall be able to use the meta-data annotation and the information encoded with the CF conventions to provide semantic annotations for the part of the semantic profile of the track data set for which there are still no annotations.

As a separate set of services this component shall be able to synchronize the semantic annotations provided to the user once he/she agrees they are correct with the underlying triple store (see the "Virtuoso" component in Figure 12) using the SPARQL1.1 update protocol.

3.1.5 WPS Metadata Service

This section shall describe the set of services that support the semantic annotation of WPS services, the overlapping between OGC standards to describe WPS services (e.g. XML profiles), and the services supporting the requirements in Section 2.1.

The set of services provided by the WPS Metadata Service component can be subdivided in two main categories: semantic annotations, kb services.

The first subset of functionalities, similarly to the previous components, allow users to semantically reconcile names of input and output parameters to a target ontology for which we can link the measured property (e.g. sea surface temperature) to a reference quantity (e.g. temperature) and a standard measuring unit (e.g. Kelvin degrees). This shall allow to restrict the application of a particular WPS service

only to some specific data sets' types and it provides a more strict constraint that the one provided by the mime type in the WPS standard.

The second set of functionalities shall implement the KB services that benefits from this semantic annotation, in detail the functionalities with id from **EO4-SEM-7** to **EO4-SEM-11**. These functionalities require that the information about the WPS services themselves shall be integrated with the controlled vocabularies managed within EO4wildlife and the application ontologies describing the known data sources and observations to produce the services.

3.1.6 Workflow Metadata Service

This section shall describe the set of services that support the semantic annotation of workflows and the services supporting the requirements in Section 2.1.3. In details, this component shall implement the functionalities with id from **EO4-SEM-12** to **EO4-SEM-15**.

This service shall be able to parse the descriptions of WPS Pipelines as described in D2.3 [3] provided with an extension of the GOC WPS service, and to represent these workflows internally as a chain of WPS services. As for the WPS services, the service shall be able to annotate the workflows definition themselves and their instantiations with ground data.

This component shall provide consistency checks when a workflow attempts to chain WPS services with incompatible I/O. This consistency check can be done at the workflow design phase, before its instantiation with ground inputs.

A separate consistency check can be provided at the instantiation phase controlling that the provided inputs' types are consistent with the description of the WPS services themselves. Moreover a separate check can be provided for workflows which attempts to model a system based on input's types which are known to be correlated.

Moreover the service shall provide, given a workflow and an initial set of data sets, the bounding box and temporal period to use for all the remaining data sets.

3.1.7 Data Retrieval Service

This section shall describe the functionalities for retrieving data sets by using the semantic annotation and using different search criteria (e.g. by using the typing information provided to the WPS I/O).

The Data retrieval component acts as a middle layer between the data management services provided by the EO4wildlife platform and the analytical components, exploiting the semantic annotations of data sets, services and workflows to cache the data prior the execution. In particular, the cache shall contain the input data sets in the format optimized for the execution. The nature of the cache is dependent on the framework used to distribute the processing burden within the services (see D3.5 for a description of the technologies used to tackle this aspect).

Once a WPS processing service has been selected for execution, the selection of the process inputs need to be performed. The WPS service defines the expected format of the inputs and outputs of the process, and the description of the service provides a description of its inputs and outputs. Several kinds of inputs can be available:

- Parameters to be edited directly by the user at execution time (input string, integer, real, ...)
- Input files previously uploaded by the user into its personal workspace, such as tracking data. This means that the file is available through a URI location on the file system.
- Input files previously made available on the EO4wildlife platform in a static way, which means that the file is available through a URI location on the file system.
This usage concerns EO data that are provided directly by EO4wildlife platform, not accessible through a connector to an external catalogue (such as bathymetry)

- Data retrieved from a catalogue search.

In this last case, the EO4wildlife data management service is used to perform the effective download of the data from the external data source.

The selection of catalogue entries through search criteria is followed by a request to access the data files associated to these criteria, sent to the data management service. EO4wildlife data management service is in charge of checking whether the required files are already stored into the platform or not, and to performed the ingestion request if not. In both cases, files are then accessible through a URI location on the platform file system.

All ingested files are stored under a common “ingestion directory”, which is accessible by GeoServer thanks to a mount point.

The path of the dedicated inputs is transmitted to GeoServer as input when executing a process. Geoserver transmits then the value of the input to the Docker container running the processing script through a variable.

The case of data uploaded by the user into his workspace is similar. The user workspaces are accessible by Geoserver thanks to a mount point and in a similar way, the path of the dedicated input are transmitted to Geoserver and then transmitted to the Docker container thanks to a variable.

In case of data format compliant with Geoserver, the data can be published into Geoserver. This case is not specific to the different kind of inputs described above but depends only on the format of the data. In this case, the data is then transmitted to Geoserver thanks to a WCS/WFS/WMS request depending on the process and data format. The data is then transmitted to the Docker container thanks to a variable.

Concerning parameters to be edited by the user at execution time, they are transmitted to the Docker container thanks to variables as well.

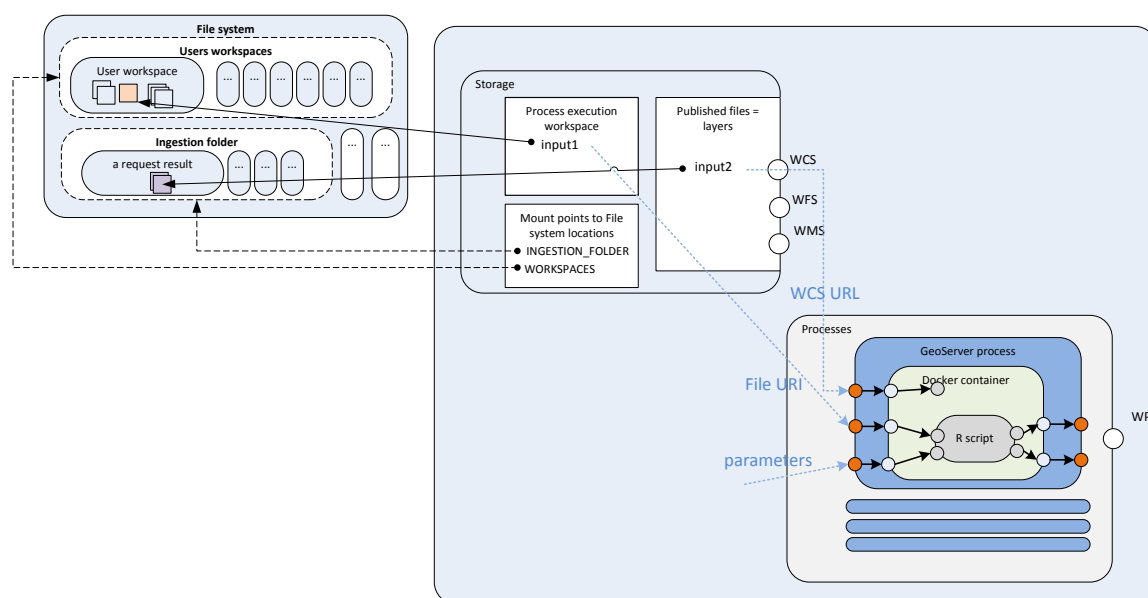


Figure 16: How data is transmitted to a processing service execution

3.2 Knowledge Base services implementation

The Knowledge Based services are developed as a set of REST APIs which interact with an instance of Virtuoso triple store via the HTTP SPARQL protocol for the management of RDF data. The implementation is provided as a composition of two Docker containers to encapsulate all dependencies.

The REST APIs are implemented in Django 1.11.1 and Python 2.7 whereas the Virtuoso container is a public image built with Virtuoso CE 7.2.4 installed.

The services are included with a UI which allows users to:

- Ingestion of NetCDF files into the system
- Semantically annotate uploaded files' variables and dimensions (see Figure 17)
- Query for available data sets using spatial bounding box and semantic annotation as input (see Figure 18)
- Ingestion of WPS Processes' XML profiles into the system (see Figure 19)

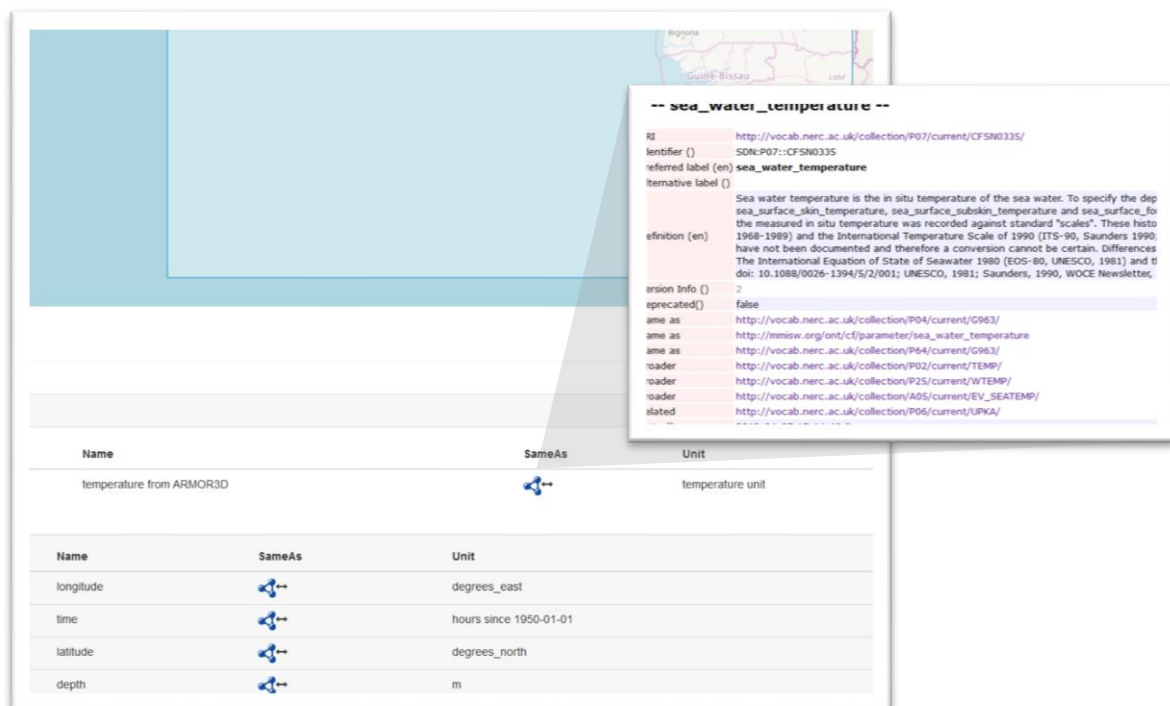


Figure 17: Semantic alignment of variable

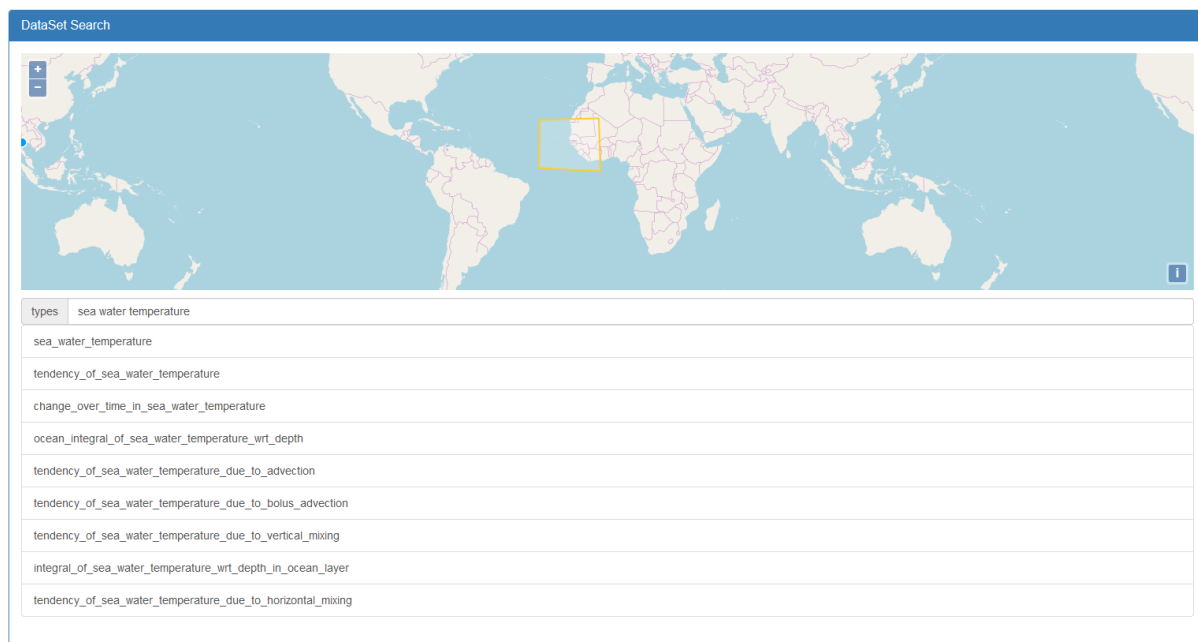


Figure 18: Spatial + Terminology based query

gc WPS process Shaded relief process


WPS process - gui/shaded_relief				
title: Shaded relief process description: Process which compute a shaded relief from a DEM file.				
I/O	Id	Name	Type	SameAs
I	SunElevation		//www.w3.org/TR/xmlschema-2/#float	n/a↔
I	VerticalExaggeration		//www.w3.org/TR/xmlschema-2/#float	n/a↔
I	InputDEM		application/netcdf	
O	ShadedRelief		application/netcdf	n/a↔

Figure 19: WPS ingestion and semantic annotation

3.3 Integration with EO4wildlife platform

This section shall describe how the KB services are meant to interact with the Semantic Services provided by the SEEED platform (e.g. Virtuoso triplestore via Linked Data API and/or SPARQL protocol)

3.3.1 CSW endpoint for catalogue

Data management service uses GeoNetwork catalogue for the management, which comes along with a CSW endpoint for catalogue access.

The OGC CSW defines the following mandatory operations:

- **GetCapabilities:** to obtain the metadata from a CSW service
- **GetRecords:** to perform a query on metadata available in catalogue, with a set of criteria
- **GetRecordById:** to get a record from its identifier
- **DescribeRecord:** to obtain the metadata structure schema

An example of a CSW Request for GetRecords with a criteria concerning the identifier:

```
http://<<eo4wl_catalogue>>/geonetwork/srv/fre/csw?outputSchema=http://www.isotc211.org/2005/gmd&ElementSetName=full&TypeName=csw:Record&NAMESPACE=xmlns(csw=http://www.opengis.net/cat/csw/2.0.2),xmlns(ogc=http://www.opengis.net/ogc),xmlns(ows=http://www.opengis.net/ows),xmlns(dc=http://purl.org/dc/elements/1.1/),xmlns(dct=http://purl.org/dc/terms/),xmlns(gml=http://www.opengis.net/gml),xmlns(xsi=http://www.w3.org/2001/XMLSchema-instance)&VERSION=2.0.2&SERVICE=CSW&MaxRecords=10&RESULTTYPE=results&outputFormat=application/xml&Constraint=(Identifier='1b0838f7')&StartPosition=1&REQUEST=GetRecords&CONSTRAINTLANGUAGE=CQL_TEXT&CONSTRAINT_LANGUAGE_VERSION=1.1.0
```

Figure 20: CSW GetRecords request via HTTP

More details on OGC Catalogue service standard can be found at:

<http://www.opengeospatial.org/standards/cat>.

3.3.2 WFS/WCS/WMS access to data

Once data is ingested into the platform, it can be accessed through WFS, WCS or WMS endpoints through GeoServer.

3.3.2.1 WCS endpoint

WCS (OGC Web Coverage Service) provides a standard interface for how to request the raster source of a geospatial image.

The following operations are available:

- **GetCapabilities:** Retrieves a list of the server's data, as well as valid WCS operations and parameters
- **DescribeCoverage:** Retrieves an XML document that fully describes the request coverages.
- **GetCoverage:** Returns a coverage as an image (JPEG, GIF, PNG, Tiff, BMP formats available) or with georeferenced formats (GeoTiff, GTopo30, ArcGrid, GZipped, and ArcGrid available)

The full reference of WCS standard is available on OGC site: <http://www.opengeospatial.org/standards/wcs>.

3.3.2.2 WFS endpoint

WFS (OGC Web Feature Service standard) defines a standard for exchanging vector data.

Operation "GetFeature" is used to return a selection of features from a data source including geometry and attribute values, with the following output formats available: GML, Shapefile, JSON, JSONP, and CSV.

The full description of WFS standard is available on OGC site: <http://www.opengeospatial.org/standards/wfs>.

3.3.2.3 WMS endpoint

WMS (OGC Web Map Service) defines a standard for requesting a geospatial map image.

The following operations are available:

- **GetCapabilities:** Retrieves metadata about the service, including supported operations and parameters, and a list of the available layers
- **GetMap:** Retrieves a map image for a specified area and content (formats supported : PNG, PNG8, JPEG, JPEG-PNG, GIF, TIFF, TIFF8, GeoTIFF, GeoTIFF8, SVG, PDF, GeoRSS, KML, KMZ, OpenLayers, UTFGrid)
- **GetFeatureInfo:** Retrieves underlying data, including geometry and attribute values, for a pixel location on a map
- **DescribeLayer:** Indicates the WFS or WCS to retrieve additional information about the layer.

The full description of WMS standard is available on OGC site:

<http://www.opengeospatial.org/standards/wms>

3.3.3 Linked Data Platform API

SparkInData semantic toolkit component provides an implementation of Linked Data Platform which provides management and storage of both RDF and non-RDF resources according to the hierarchical data structure defined in LinkedDataPlatform standard [43].

Storage is performed in two different ways for RDF or non-RDF resources:

- For RDF resource (expressed as triplets): storage is performed into Virtuoso triple store
- For non-RDF resource (binary / text file): storage of a “pointer” is performed inside Virtuoso triple store and effective storage of the resource is done into a NoSQL Database (MongoDB) available in SparkInData components.

In EO4wildlife context, Virtuoso is used as a triple store for RDF resource storage and is accessible through Linked Data Platform API for RDF resource management and SPARQL queries.

The LDP API provided by SparkInData Semantic toolkit provides the following functionalities through a REST API:

- Upload a data to the triple store
- Get a resource from the server
- Update a resource (partial or full update)
- Delete a resource
- Retrieve meta-information associated to a resource
- Get the available options of a resource
- Handling of SparQL queries

4 Conclusion

The present document outlined the architecture of the Knowledge Base services and its integration with the overall EO4wildlife system. The deliverable described the overall aims and objective of using semantic services in support of the data mining and fusion analytics and it specified how the services will support them.

The main ontology design decisions have been outlined in reference to the functionalities to be implemented and backed up by a literature review of relevant approaches.

The results of this document will guide the development phase of the knowledge base services and it will provide guidelines for the overall EO4wildlife platform development on how to extend the capabilities provided by available OGC standards.

This deliverable is tightly connected with the deliverable D3.3 “Big data connectors and catalogue service” [4] with which share some of the decisions in describing the metadata about different data sources.

References

- [1] G. Weller, E. Lambert, and J.-M. Zigna, 'D1.1 Use Case scenarios v1', Deliverable of the EO4wildlife project
- [2] A. Haugommard, F. Martin, and D. Rodera, 'D2.1 System architecture and operational scenarios v1', Deliverable of the EO4wildlife project
- [3] A. Haugommard, 'D2.3 External interface for data discovery and processing v1', Deliverable of the EO4wildlife project
- [4] J.-M. Zigna, 'D3.3 Big data connectors and catalogue service v1', Deliverable of the EO4wildlife project.
- [5] G. Correndo, G. Veres, and B. Arbab-Zavar, 'D3.5 Data Mining and High Level Data Fusion Services v1', Deliverable of the EO4wildlife project
- [6] B. Li, 'Sustainable Value and Generativity in the Ecological Metadata Language (EML) Platform: Toward New Knowledge and Investigations', in *2014 47th Hawaii International Conference on System Sciences*, 2014, pp. 3533–3542
- [7] J. Madin, S. Bowers, M. Schildhauer, S. Krivov, D. Pennington, and F. Villa, 'An ontology for describing and synthesizing ecological observation data', *Ecological Informatics*, vol. 2, no. 3, pp. 279–296, Oct. 2007
- [8] S. Cox, 'Observations and Measurements', Open GIS Consortium Inc., 03–022r3, 2003
- [9] M. G. Hidas *et al.*, 'Information infrastructure for Australia's Integrated Marine Observing System', *Earth Sci Inform*, pp. 1–10, May 2016
- [10] 'Research Vocabularies Australia', RVA. [Online]. Available: <https://vocabs.ands.org.au/>. [Accessed: 21-Oct-2016]
- [11] N. Car, 'Australian Government Linked Data Working Group home page', 23-Jul-2015. [Online]. Available: <http://environment.data.gov.au/>. [Accessed: 21-Oct-2016]
- [12] A. Miles, B. Matthews, M. Wilson, and D. Brickley, 'SKOS Core: Simple knowledge organisation for the Web', *International Conference on Dublin Core and Metadata Applications*, vol. 0, no. 0, pp. 3–10, Sep. 2005
- [13] M. Blumenthal, M. Bell, J. del Corral, R. Cousin, and I. Khomyakov, 'IRI Data Library: enhancing accessibility of climate knowledge', *Earth Perspectives*, vol. 1, no. 1, p. 19, 2014
- [14] 'IRI/LDEO Climate Data Library'. [Online]. Available: <http://iridl.ldeo.columbia.edu/>. [Accessed: 21-Oct-2016]
- [15] J. Yu, N. J. Car, A. Leadbetter, B. A. Simons, and S. J. Cox, 'Towards linked data conventions for delivery of environmental data using netCDF', in *International Symposium on Environmental Software Systems*, 2015, pp. 102–112
- [16] 'European Commission : CORDIS : Projects & Results Service : Final Report Summary - SEADATANET II (SeaDataNet II: Pan-European infrastructure for ocean and marine data management)'. [Online]. Available: http://cordis.europa.eu/result/rcn/181547_en.html. [Accessed: 21-Oct-2016]
- [17] 'SeaDataNet project'. [Online]. Available: <http://www.seadatanet.org/>. [Accessed: 21-Oct-2016]
- [18] S. Cox, K. Mills, and F. Tan, 'Vocabulary services to support scientific data interoperability', in *EGU General Assembly Conference Abstracts*, 2013, vol. 15, p. 1143
- [19] A. Leadbetter, 'NERC Vocabulary Server version 2.0'
- [20] N. DiGiuseppe, L. C. Pouchard, and N. F. Noy, 'SWEET ontology coverage for earth system sciences', *Earth Sci Inform*, vol. 7, no. 4, pp. 249–264, Jan. 2014
- [21] 'Semantic Web for Earth and Environmental Technology (SWEET)'. [Online]. Available: <https://sweet.jpl.nasa.gov/>

- [22] 'CF Conventions Home Page'. [Online]. Available: <http://cfconventions.org/>. [Accessed: 21-Oct-2016]
- [23] 'MMI Ontology Registry and Repository'. [Online]. Available: <http://mmisw.org/orr/>. [Accessed: 21-Oct-2016]
- [24] M. Compton *et al.*, 'The SSN ontology of the W3C semantic sensor network incubator group', *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, pp. 25–32, Dec. 2012
- [25] M. Perry and J. Herring, 'OGC GeoSPARQL-A geographic query language for RDF data', *OGC Implementation Standard, ref: OGC*, 2011
- [26] 'Geosemantics DWG | OGC'. [Online]. Available: <http://www.opengeospatial.org/projects/groups/semantics>. [Accessed: 21-Oct-2016]
- [27] R. Cyganiak, D. Reynolds, and J. Tennison, 'The RDF data cube vocabulary', *W3C Recommendation (January 2014)*, 2013
- [28] 'SDMX standards. Information Model UML Conceptual Design', Jul. 2011
- [29] J. D. Blower and M. Riechert, 'Coverages, JSON-LD and RDF Data Cubes', presented at the Workshop on Spatial Data on the Web, 2016
- [30] T. Lebo *et al.*, 'PROV-O: The PROV Ontology, 2013', *W3C Recommendation*, 2013
- [31] L. Moreau *et al.*, 'The Open Provenance Model core specification (v1.1)', *Future Generation Computer Systems*, vol. 27, no. 6, pp. 743–756, 2011
- [32] D. T. Michaelides, R. Parker, C. Charlton, W. J. Browne, and L. Moreau, 'Intermediate Notation for Provenance and Workflow Reproducibility', in *Provenance and Annotation of Data and Processes*, M. Mattoso and B. Glavic, Eds. Springer International Publishing, 2016, pp. 83–94
- [33] B. Domenico, 'OGC Network Common Data Form (NetCDF) Core Encoding Standard version 1.0', *Open Geospatial Consortium Inc.. document*, vol. 10, p. e090r3, 2011
- [34] R. Cyganiak, D. Wood, and M. Lanthaler, 'RDF 1.1 concepts and abstract syntax', *W3C Recommendation*, vol. 25, pp. 1–8, 2014
- [35] J. D. Fernández, M. A. Martínez-Prieto, C. Gutiérrez, A. Polleres, and M. Arias, 'Binary RDF representation for publication and exchange (HDT)', *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 19, pp. 22–41, 2013
- [36] F. Royer and M. Lutcavage, 'Positioning pelagic fish from sunrise and sunset times: error assessment and improvement through constrained, robust modeling', *Rev.: Methods Technol. Fish Biol. Fish*, vol. 9, pp. 323–342, 2009
- [37] J.-M. Zigna, 'D3.4 Big data connectors and catalogue service v2', Deliverable of the EO4wildlife project
- [38] 'CF Standard Names'. [Online]. Available: <http://cfconventions.org/Data/cf-standard-names/35/build/cf-standard-name-table.html>. [Accessed: 02-Nov-2016]
- [39] R. Verborgh and M. De Wilde, *Using OpenRefine*. Packt Publishing Ltd, 2013
- [40] 'Reconciliation Service API'. [Online]. Available: <https://github.com/OpenRefine/OpenRefine/wiki/Reconciliation-Service-API>
- [41] 'The BirdLife checklist of the birds of the world: Version 8.', *BirdLife International (2015) The BirdLife checklist of the birds of the world: Version 8*. [Online]. Available: http://www.birdlife.org/datazone/userfiles/file/Species/Taxonomy/BirdLife_Checklist_Version_80.zip
- [42] 'Reptile Database data sets for download'. [Online]. Available: <http://www.reptile-database.org/data/>. [Accessed: 28-Oct-2016]
- [43] 'Linked Data Platform 1.0'. [Online]. Available: <https://www.w3.org/TR/ldp/>. [Accessed: 10-Nov-2016]