EcoPortal Documentation

Version 1.0

Lecce, 10 May 2021

Authors:

Pierfrancesco Tommasino, National Research Council, Research Institute on Terrestrial Ecosystems, Rome, Italy; pierfrancesco.tommasino@cnr.it

Lucia Vaira, LifeWatch ERIC, Service Centre, Lecce, Italy; lucia.vaira@lifewatch.eu

Nicola Fiore, LifeWatch ERIC, Service Centre, Lecce, Italy; nicola.fiore@lifewatch.eu

Xeni Kechagioglou, LifeWatch ERIC, Service Centre, Lecce, Italy; xeni.kechagioglou@lifewatch.eu

Ilaria Rosati, National Research Council, Research Institute on Terrestrial Ecosystems, Rome, Italy; ilaria.rosati@cnr.it

www.lifewatch.eu

Table of contents

1. About EcoPortal

What is EcoPortal and who developed it?
EcoPortal is a repository for semantic resources in the ecological domain, developed by LifeWatch ERIC, for different kinds of stakeholders (researchers, vocabularies managers, data managers, ontologists, etc.). EcoPortal supports the community in the creation, management, mapping and alignment of its semantic resources and subsequently also of its data.
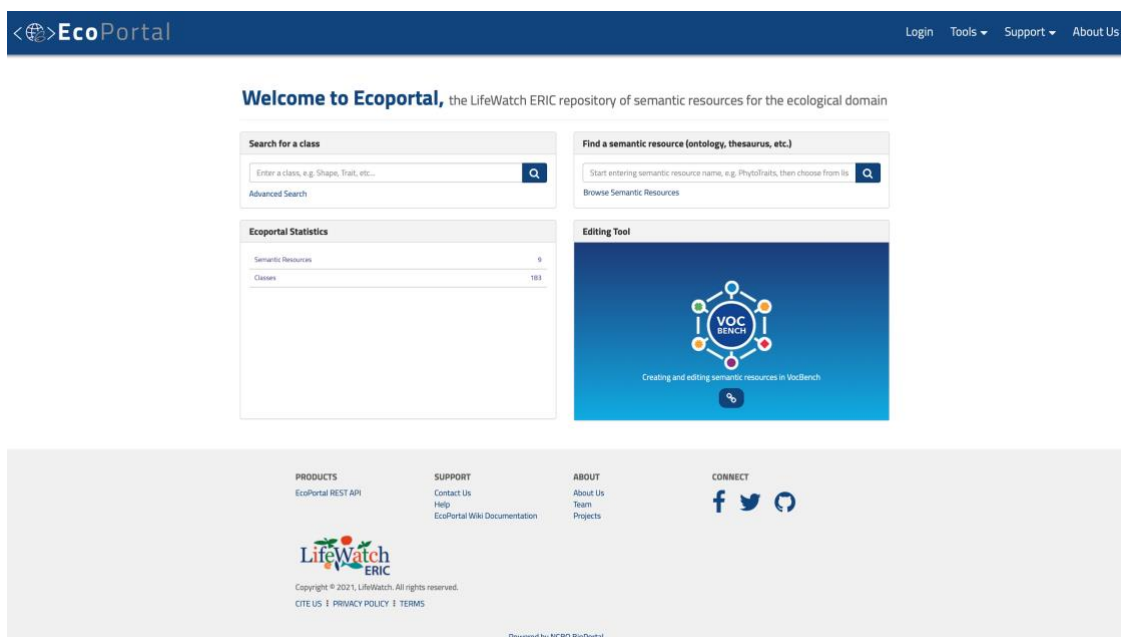
Why EcoPortal and how does it work?

In the last decade, ecological research groups and infrastructures have contributed to the production of core and domain ontologies, as well as controlled vocabularies to increase data discovery, integration and interoperability. To make the discovery and use of all these semantic resources easier, LifeWatch ERIC developed EcoPortal which acts not only as a single platform, allowing users to identify and select semantic resources for specific tasks, but also offers generic services to maximise the efficiency of search, annotation and other scientific data management processes.

The EcoPortal Home Page (see Figure 1) is constituted by:

- a header that includes the logo on the left and the navigation bar to the right
- two calls to actions and some statistical information as plain content
- a footer that includes all the support information.

NavBar Ecoportal

The navigation bar on the upper right side of the web page includes four menu items to navigate within EcoPortal:

Figure 1: EcoPortal Home Page

1. Login - to access the login/registration form. If you are already logged in, your username will be displayed here
2. Tools - to access the list of tools provided by EcoPortal, which are:
   - Browse
   - Search
   - Publish
   - Annotator
   - Mappings
   - Recommender
   - VocBench
3. Support - to ask for support in relation to questions, error reports, feedback, etc. by means of a user-friendly contact form
4. About Us - information in relation to who we are, what LifeWatch ERIC is and what EcoPortal is.

The Home Page contains two sections where you can quickly view or query the available resources. In particular you can:
- quickly search for terms by entering a class name into the "Search for a class" field in the form to be redirected to the page showing the results
- start entering free text in the "Find a semantic resource" field in order to quickly find the resource you are interested in, by using the autocompletion functionality and then selecting the summary page.

Moreover, you can also see the EcoPortal statistics such as number of published resources, number of classes, etc.

The footer includes all EcoPortal support links, divided into four sections:
1. Products:
   - EcoPortal REST API: to access the API Documentation of EcoPortal.
2. Support:
   - Contact Us: to ask for support in relation to questions, error reports, feedback, etc. by means of a user-friendly contact form
   - Help: to access information on how to get started and how to find documentation
   - EcoPortal Wiki Documentation: to access this online documentation.
3. About:
   - About us: to read information in relation to who we are, what LifeWatch ERIC is and what EcoPortal is

- Team: to see the EcoPortal team members
- Project: to view selected projects which make use of EcoPortal resources. To add your project to this list, click the "Create New Project" button. You must be logged into EcoPortal to create a new project
4. Connect - shortcuts to access our social media channels.

2. EcoTools

EcoPortal provides access to ecological semantic resources and tools for working with them:

- **browse** the semantic resources
- **search** for a term across multiple semantic resources
- **publish** a semantic resource
- **browse mappings** among terms in different semantic resources
- **receive recommendations** on which semantic resources are most relevant for a given corpus
- **annotate text** with terms from semantic resources
- **browse a selection of projects** that use EcoPortal resources
- **create and edit semantic resources** with VocBench Editor**.**

All information available through the EcoPortal website is also available through the EcoPortal Web service REST API. Please see REST API documentation for more information.

2.1 Browse

The Browser tool allows you to search semantic resources. When you open the browser page to find a semantic resource (see Figure 2), you have two options:

1. you can go through the list and select the semantic resource to see more detailed information
2. you can start typing the semantic resource name or abbreviation in the "Search…" field.

In both cases, you can further refine your search by using the filters provided on the left-hand side of the page. To add a filter to your search, click the checkbox or filter label. Upon selecting the filter, the search will be updated to show results which contain the selected filter. Repeat the process to add additional filters.

Moreover, you can sort the list by choosing the desired criteria (upper right side of the list of resource):

- Popular: number of visits to the semantic resource
- Size: number of classes or concepts
- Projects: number of projects where the semantic resource is used
- Notes: number of notes issued for that semantic resource.

Figure 2: Browser page

Furthermore, once logged into EcoPortal, in the "Account Settings" section, you can select only the semantic resources that you need to see, without viewing others (see Figure 3).
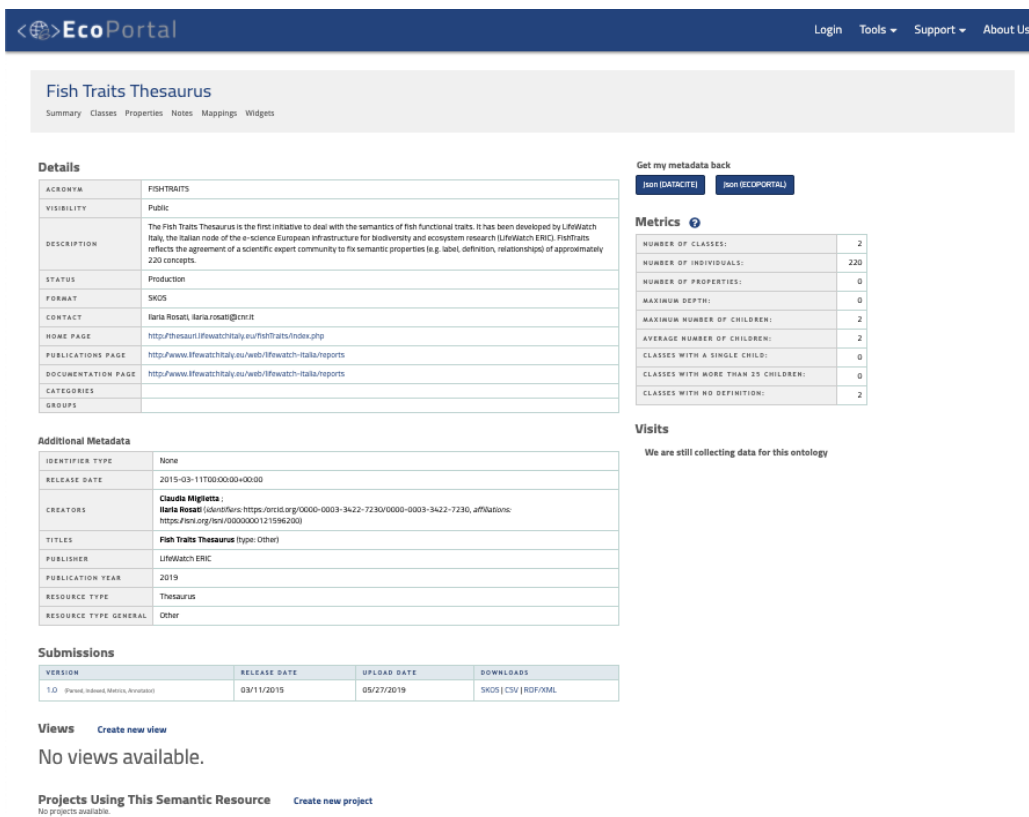
Figure 3: Custom set of semantic resources

To access semantic resources programmatically, see our Resources REST API documentation (then take a look under Media Types).

You will be able to see semantic resources that are marked "Public" because submitters have the facility to control who can browse the terms in their semantic resources by marking them with "Public" or "Private" visibility. In order to browse the terms in these private resources you have to be logged into EcoPortal and have access to the semantic resource. In particular:

- for private semantic resources, the submitter directly controls the access. In order to gain access to a specific private semantic resource you have to contact the submitter and you must be registered in EcoPortal. Private semantic resources will not be indexed in the search

- for public semantic resource: the resources are accessible and can be viewed by all users even if they are not logged in.

Figure 4: The summary page of semantic resources

In the same browser page, you will find the "Submit New Semantic Resource" button on the upper-left part of the page, but you need to be logged in to EcoPortal to perform this operation. Once you click the button, you have to select the required fields (marked with a red *) as described in "How to publish a semantic resource".

Once opened the selected semantic resource, the summary page will show the different areas (see Figure 4):

a) menu area

b) details area

c) download area

d) metrics area

e) submission area

f) views area

g) project area.

a) Menu area
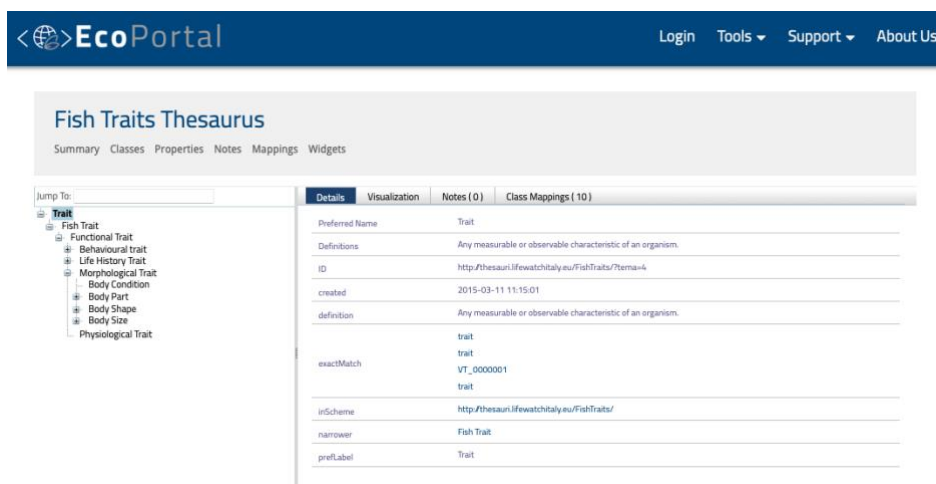
The menu area allows you to:

    1.1 View classes

    1.2 View properties

    1.3 View and edit notes

    1.4 View and edit mappings

    1.5 Select a widget.

1.1 Classes

In the summary page of the semantic resource, when clicking on Tab Classes, you will see:

- on the top of the left column a search field for classes and concepts discovery; below the navigation tree of the concepts or interactive classes, whether it is ontology or controlled vocabulary
- on the right column an area with Details, Visualization, Notes and Mappings related to the selected class/concept:
  - Details: tab showing properties and relationships associated with the selected class/concept (see Figure 5). The properties shown will differ according to the representation model of the resource (SKOS; OWL, UMLS; OBO). The custom properties are also displayed, for example: ID, created, definition, etc.

Figure 5: View classes of a specific semantic resource - details

- Visualisation: EcoPortal displays semantic resources as a graph in which the

resource concepts are represented by nodes and the relationships among concepts are displayed as edges between the nodes (see Figure 6). By default, the most basic directional relationships (parent/child relationships) are identified by a solid navy-blue line with an arrow pointing to the child concept. Mappings are identified by a solid light-grey line. However, depending on the construction of your semantic resource, EcoPortal can display other types of relationships.
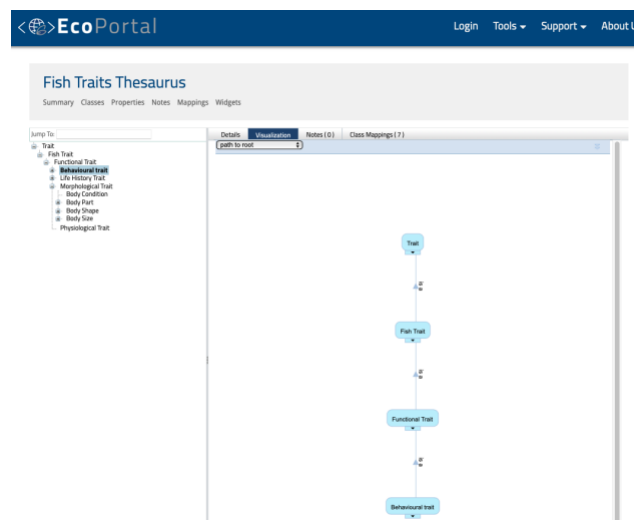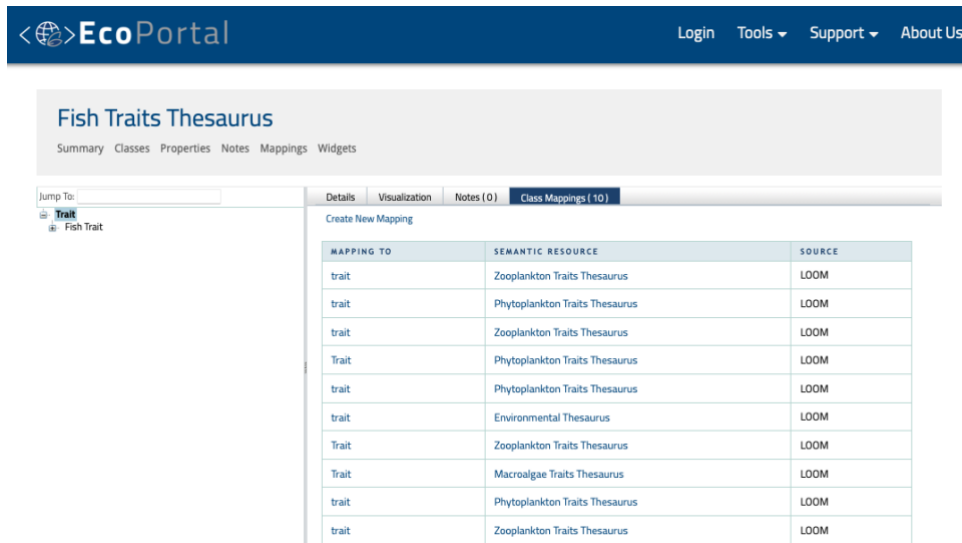
Figure 6: View classes of a specific semantic resource - visualization

- Notes: tab showing all the notes issued by EcoPortal users in relation to the selected class/concept. You must be logged in to be able to issue generic comments or proposals regarding new classes, new relationships or to change properties.
    - Class Mappings: tab showing all the mappings associated with the selected class/concept (see Figure 7). For each mapping you can see:
    - mapping to: the name of the selected class/concept mapped to the class/concept of another resource
    - semantic resource: the semantic resource of the mapped class/concept
    - source: the mapping algorithm.

Figure 7: View classes of a specific semantic resource - visualization

1.2 Properties

In the summary page of the selected semantic resource, when clicking on Properties Tab, you can see all the properties used within the selected semantic resource (see Figure 8). The same

Figure 8: View properties of a specific semantic resource

as for the Classes Tab, the interactive property tree is shown on the left, while the property details are shown on the right.

1.3 Notes

In the Notes tab you can see all the comments or proposals issued by registered EcoPortal users (see Figure 9). You can add comments or proposals only if you are registered and logged in by clicking on the "Add Comment" and "Add Proposal" buttons.
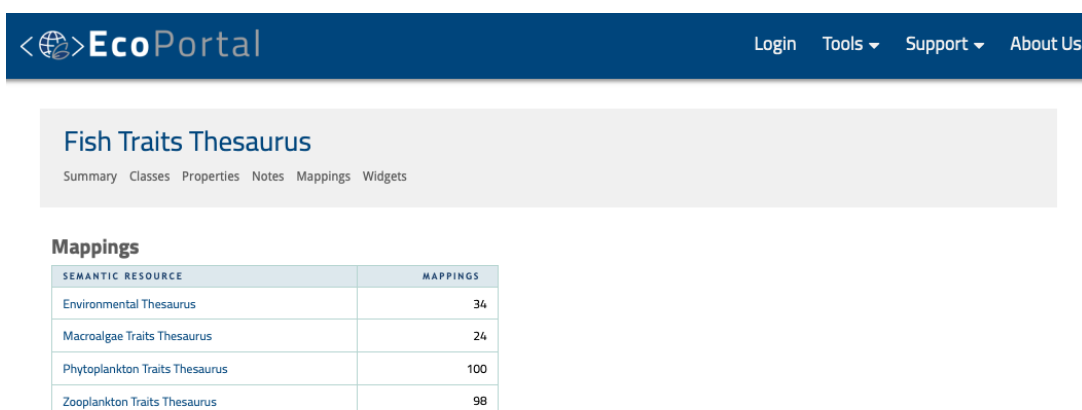
Figure 9: View notes of a specific semantic resource

- **Add a new comment:** you can send a comment to the owner of the resource by specifying the subject and writing a free text comment
- **Add a new proposal:** you can send a proposal to the owner of the resource to modify or create new relationships or classes / concepts
- **Subscribe/Unsubscribe to notes emails:** you can subscribe to receive emails whenever comments or proposals from other users are provided by clicking on the "Subscribe to notes emails" link. You can manage your subscriptions on the "Account Settings" page and click on "Unsubscribe" link to stop receiving emails.

1.4 Mappings

In the Mappings Tab you can see all the mappings of the selected semantic resource (see Figure 10). The table includes two columns: Semantic Resource and Mappings. The first identifies the semantic resource where mappings are found, and the second shows the number of mappings. You can click on a semantic resource to see all the mappings found.

Figure 10: View mappings of a specific semantic resource
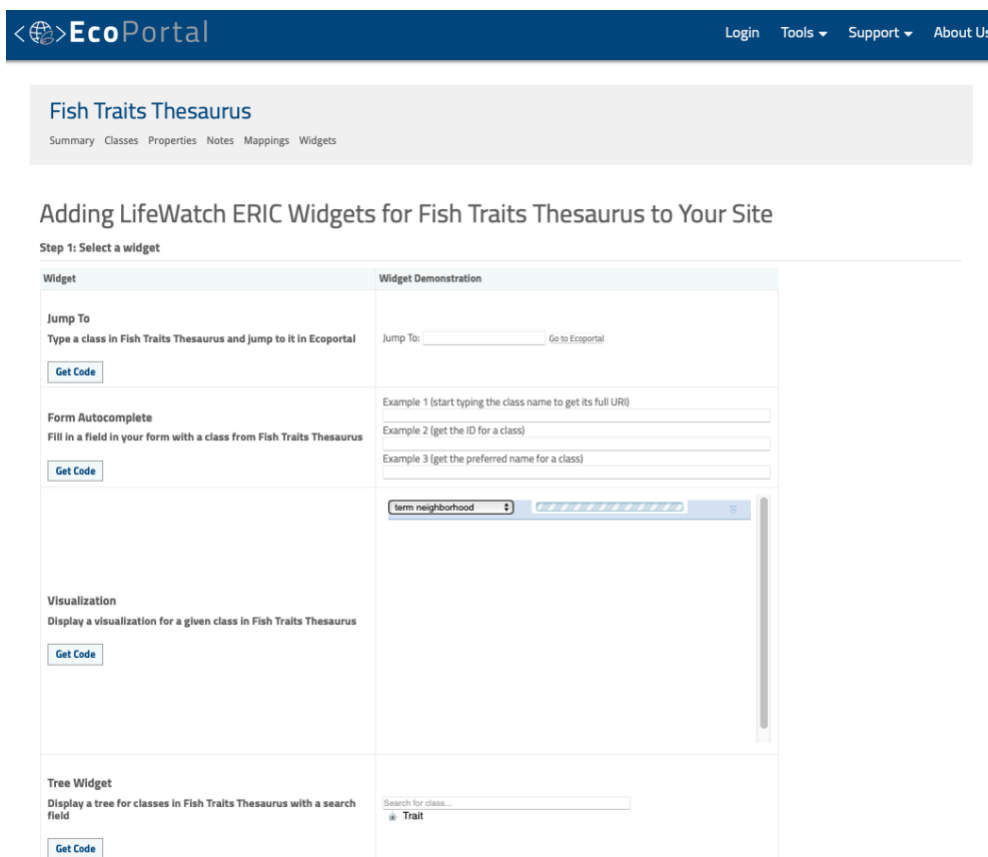
1.5 Widgets

In the Widgets Tab you can see different types of widgets that you can use within a web page (see Figure 11).

The types of widgets available within EcoPortal are:

- Jump To: type a class in the selected semantic resource and jump to it in EcoPortal;
- Form Autocomplete: fill in a field in your form with a class from the selected semantic resource;
- Visualization: display a visualization for a given class/concept in the selected semantic resource
- Tree Widget: display a tree for classes in the selected semantic resource with a search field.

You can use the widgets only if you are registered in EcoPortal with an associated API-KEY.

Figure 11: View widgets of a specific semantic resource

b) Details area

In the detail area of the summary page (see Figure 4) you can view the EcoPortal metadata schema for the selected semantic resource that are taken from the metadata schema provided by BioPortal:

ACRONYM	Semantic resource acronym.
VISIBILITY	Resource visibility: Private or Public. You will only be able to read Private resources if you have previously requested access to them.
DESCRIPTION	Description of the resource.
STATUS	Resource release status: alpha, pre-production. production, etc.
FORMAT	Resource format: SKOS, OWL, UMLS, OBO.
CONTACT	Contact information for the resource authors.
HOME PAGE	Homepage for the resource: URL.
PUBLICATIONS PAGE	Resource publication page: URL.
DOCUMENTATION PAGE	Resource documentation page: URL.
CATEGORIES	Categories to which the resource belongs.
GROUPS	Groups to which the resource belongs.

In the **Additional Metadata** section, instead, you will find the DataCite Metadata Schema v4.3 needed to obtain the DOI (Digital Object Identifier):

IDENTIFIER	The DOI (Digital Object Identifier) or other identifier (e.g., PID) of the resource if it exists with the format "10.1234/foo".
IDENTIFIER TYPE	Controlled List Value: DOI, Other.
RELEASE DATE	Date on which the resource was published / released on EcoPortal.
CREATORS	The main researchers/institution/organization involved in producing the resource, in priority order. May be a corporate/institutional or personal name.
TITLES	The name or title by which a resource is known. The title is a controlled list of values: AlternativeTitle, Subtitle, TranslatedTitle, Other.
PUBLISHER	The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. In EcoPortal, if a DOI is requested, the publisher will always be: LifeWatch ERIC.

PUBLICATION YEAR	The year when the data was or will be made publicly available.
RESOURCE TYPE	The description of the resource. E.g., Controlled Vocabulary, Ontology.
RESOURCE TYPE GENERAL	The general type of a resource.

c) Download area

On the right side of the summary page and above the metrics area there are two buttons for downloading metadata (see Figure 12):

- Json (DATACITE) button provides the DataCite metadata
- Json (ECOPORTAL) button provides all EcoPortal metadata.

Figure 12: Buttons to download metadata

d) Metrics area

This section describes the metrics that EcoPortal computes for the semantic resources in its repository (see Figure 13). EcoPortal calculates metrics when the semantic resource is uploaded and will store them as part of the BioPortal Ontology Metadata. There are two groups of metrics:

- statistical metrics
- quality-control and quality-assurance metrics.

Some metrics are meaningful only for semantic resources in a specific representation language (e.g., there are no individuals to count in the ontologies in OBO format).

Figure 13: Metrics area of the summary page of a semantic resource

Statistical Metrics are calculated on each semantic resource:

- number of classes: the number of named (not anonymous) classes in the semantic resource
- number of individuals: the number of individuals in the semantic resource
- number of properties: the number of properties or slots in the semantic resource

- maximum depth: the maximum depth of the hierarchy tree:
    - for OWL, RDFS, Protege, and RRF ontologies, we consider only the "is-a" relationship as a hierarchical relationship;
    - for OBO format ontologies, we consider the following relationships as hierarchical relationships: "is-a", "has-part", "inverse" and "develops-from"
- maximum number of children: the maximum number of children in the semantic resource
- average number of children: the average number of children at one level in the tree.

Quality-Control and Quality-Assurance Metrics provide some indication of the quality of the semantic resource and help semantic resource authors to improve their quality. To improve their usefulness, we limit the size of lists for any one of these metrics to 200. Semantic resources with more than 200 classes in a category will still have the total number of classes in that category counted, but no list will be available. Please notice that:

- classes with a single child: a list of classes that have only one subclass in the is-a hierarchy. While technically there is no problem in having only one subclass, this situation often indicates that either the hierarchy is under-specified, or the distinction between the class and the subclass is not appropriate
- classes with more than 25 children: a list of classes that have more than direct subclasses. A class that has more than 25 subclasses is a candidate for which additional distinctions and categorization are needed
- classes with no definition: a list of classes that have no value for the definition property. For ontologies in OBO and RRF formats, the property for definition is part of the language. For OWL ontologies, the authors specify this property as part of the ontology metadata (the default is "skos:definition").

For more details about the metrics generation, please visit:

https://www.bioontology.org/wiki/Ontology_Metrics.

EcoPortal users will be able to access semantic resource metrics in two ways:

- through the EcoPortal user interface, in the metrics area of the summary page (see example here)
- through a dedicated REST service that returns a JSON.

In relation to the REST access, for example, the following REST service can be used:

http://ecoportal.lifewatch.eu:8080/ontologies/FISHTRAITS/metrics?apikey=YourAPIKey

The service returns a JSON that contains the version id for the semantic resource and the values for the metrics:

```json
{
  "id": "http://193.204.79.110:8080/ontologies/FISHTRAITS/submissions/1/metrics",
  "classes": 2,
```

```
    "individuals": 220,
    "created": "2019-05-27T07:00:40-07:00",
    "properties": 0,
    "maxDepth": 0,
    "maxChildCount": 2,
    "averageChildCount": 2,
    "classesWithOneChild": 0,
    "classesWithMoreThan25Children": 0,
    "classesWithNoDefinition": 2,
    "submission": [
          "http://193.204.79.110:8080/ontologies/FISHTRAITS/submissions/1"
     ],
    "@id": "http://193.204.79.110:8080/ontologies/FISHTRAITS/submissions/1/metrics",
    "@type": "http://data.bioontology.org/metadata/Metrics",
    "links": {
          "ontology": "http://193.204.79.110:8080/ontologies/FISHTRAITS",
          "submission": "http://193.204.79.110:8080/ontologies/FISHTRAITS/submissions/1",
    "@context": {
          "ontology": "http://data.bioontology.org/metadata/Ontology",
          "submission": "http://data.bioontology.org/metadata/OntologySubmission"
     }
    },
    "@context": {
          "@vocab": "http://data.bioontology.org/metadata/"
     }
    }
```

If a given semantic resource has more than 200 classes identified for any single metric, then that metric will provide an error message in place of the class list. Two states are available:

- **limitpassed:** it means that the 200-class limit was reached and passed. When this error presents, it is followed by the total number of classes in the semantic resource that matches this metric
- **alltriggered:** it means that every class in the semantic resource matched this metric.

e) Submission area

VERSION	RELEASE DATE	UPLOAD DATE	DOWNLOADS
1.2 (Parsed, Indexed, Metrics, Annotator)	12/02/2020	12/10/2020	SKOS	CSV	RDF/XML	Diff
1.1 (Archived)	11/02/2020	12/02/2020	SKOS	Diff
1.0 (Archived)	10/01/2020	10/10/2020	SKOS

In the submission area all submissions made for that resource are available. The table includes four columns:

- Submission: shows the version loaded in EcoPortal with the labels next to it. The labels assigned automatically by the portal are parsed (the resource has been parsed), indexed (the resource has been indexed), metrics (the metrics for that resource have been calculated) and annotator (the resource is ready to annotate some text via API or the Annotator tools). EcoPortal indexes the latest submission. The previous submissions will show the "Archived" label, which means that submission will no longer be published
- Release date: shows the date (mm/dd/yy) when the new resource was released (manually set by the user)
- Upload date: shows the date (mm/dd/yy) when the resource was published on EcoPortal for the first time (automatically set by the system)
- Downloads: permits download of the resource in different formats SKOS, OBO, UMLS, OWL, CSV, RDF/XML, depending on the format of the resource. Furthermore, when more submissions are made for a resource, the system allows an automatic download report (Diff) of the differences between the submitted versions.

6. Views area
In this area you can see the views for the selected semantic resource or create a new view. A view provides a manageable portion of a larger semantic resource for localised applications and users.
If you are creating a resource as a view, it means that:

- it allows the precise extraction of sub-resources of a larger resource
- it allows for localised personalization and use of the larger portion of a resource
- it enables the interoperability between large semantic resources and applications.

Each displayed view is linked for an easy consultation.
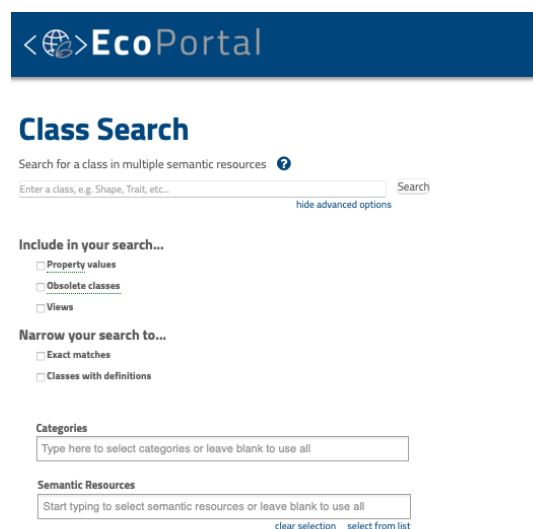
7. Project area
In this area you can see the projects using the selected semantic resource or create a new project. Each project displayed is linked for easy consultation.

2.2 Search
The Search tool allows you to find a term across semantic resources by entering free text in the "Search…" field and clicking on the "Search" button. The system looks for matches in the term name, synonyms, term IDs, and in property values. You can use the advanced search options for including in your search (see Figure 14):

- Property Values: named association between two entities. Examples are "definition" (a relationship between a class and some text) and "part of" (a relationship between two classes)
- Obsolete Classes: a class that the authors of the semantic resource have flagged as being obsolete and which they recommend that people not use. These classes are often left in semantic resources (rather than removing them entirely) so that existing systems that depend on them will continue to function
- Semantic Resource Views: search only ontological views

Moreover, you can narrow your search by selecting "exact matches" and/or "classes with definitions". Finally, you can type in the "Categories" field to select one or more categories and/or you can also select a set of semantic resources from the "select from list" field. The search will be refined according to only the categories and semantic resources selected.

Figure 14: The search tool

Because of performance concerns, the search returns only the top 100 matches. The search result will display the matched terms and their associated semantic resource. For each search result you can also see (links in green):

- details: a window with the associated properties of the searched term
- visualize: the graph with the nodes of the terms with the path to the root
- <number> more from this semantic resource: other terms, present in the semantic resource of the result, are shown.

To search programmatically see our Search REST API documentation.

2.3 Publish

The Publish tool allows you to publish a new semantic resource. For details see the dedicated section of the documentation.

2.4 Annotator

The Annotator tool allows you to generate annotations by entering or pasting free text in the box and clicking on the "Get Annotations" button (see Figure 15). The system matches words in the text to terms in the semantic resources by doing an exact string comparison (a "direct" match) between the text and semantic resource term names, synonyms, and IDs.
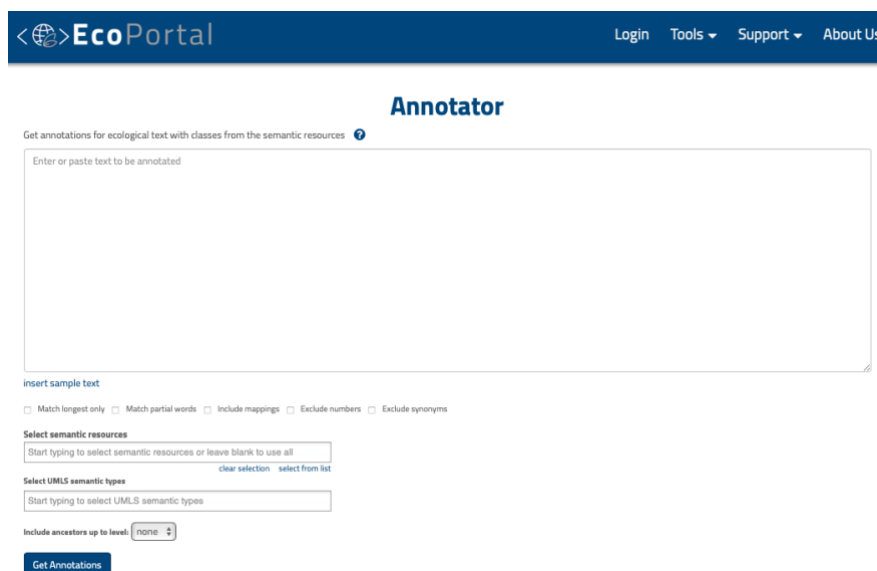
Figure 15: The annotator tool

In addition to the direct matches, the user may expand the set of matches by including matches from mapped terms and from hierarchical expansion. For most semantic resources (OWL and UMLS RRF), the system performs the hierarchical expansion on the superclass ("is-a") relationship. For OBO semantic resources the hierarchical expansion also includes the part-of relationship. Use the "number of levels" field to control the number of levels up the hierarchy for which the system will return terms for a given match.
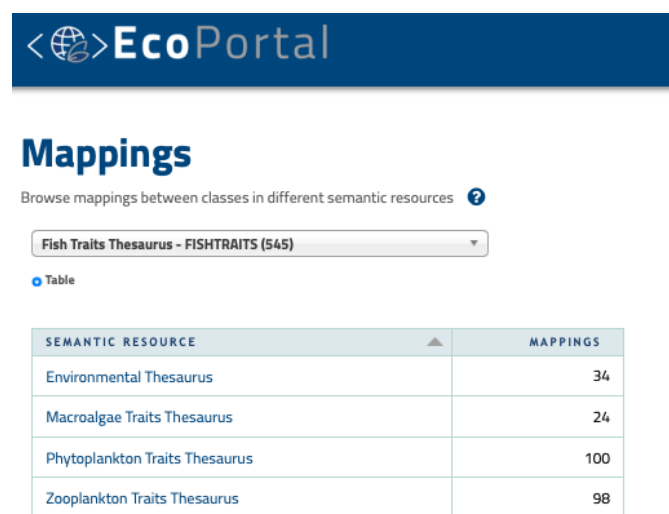
To generate annotations programmatically, see the REST API documentation. For more details about the workings of the Annotator, please visit Shah et al BMC Bioinformatics. 2009 Sep 17;10 Suppl 9:S14.

2.5 Mappings

The Mappings tool allows you to associate two or more terms in different semantic resources.

This association typically, but not always, represents a degree of similarity between the terms. The author of the mapping defines the semantics of a particular mapping. It is also usual for a mapping to be bi-directional, but again, this is not required.

To browse mappings for a semantic resource, select the semantic resource from the drop-down menu. You will then see a table showing all semantic resources for which at least one mapping exists between that semantic resource and the semantic resource you selected (see Figure 16). Select a semantic resource from this table to browse the mappings between the two semantic resources. To access mappings programmatically, please see our Mappings REST API documentation.

Figure 16: The mapping tool

Mapping Sources

Mappings are generated by a LOOM algorithm automatically based on close lexical match between preferred names of terms or a preferred name and a synonym. The lexical match involves removing white-spaces and punctuation from labels. Any labels with no more than 3 characters are excluded. These mappings are represented by the mapping relationship "skos:closeMatch".

Mapping for terms with the same URI from different semantic resources are represented by the mapping relationship "skos:exactMatch".

Mappings between semantic resource terms related by an OBO xref property. are represented by the mapping relationship "skos:relatedMatch". See also OBO mappings.

Mapping Relationships

There are mappings of the following types in EcoPortal:

- **identical,** the same: owl:sameAs

- ○ Definition (from OWL): "owl:sameAs" is used to state that two URI references refer to the same individual
- **related** (but not necessarily similar):rdfs:seeAlso
  - ○ Definition (from RDFS): the property "rdfs:seeAlso" specifies a resource that might provide additional information about the subject resource
- **related match**:skos:relatedMatch
  - ○ Definition (from SKOS-REFERENCE): the property "skos:relatedMatch" is used to state an associative mapping link between two concepts; "skos:relatedMatch" is a sub-property of "skos:mappingRelation"
  - ○ Symmetry "skos:relatedMatch" is an instance of "owl:SymmetricProperty"; for bidirectional mappings between "" and "" that are "rdf:type skos:Concept":
    - ▪ "skos:relatedMatch"
    - ▪ "skos:relatedMatch"
- **close match**:skos:closeMatch
  - ○ Definition (from SKOS-REFERENCE): the property "skos:closeMatch" is used to link two concepts that are sufficiently similar that they can be used interchangeably in some information retrieval applications. In order to avoid the possibility of "compound errors" when combining mappings across more than two concept schemes, "skos:closeMatch" is not declared to be a transitive property. The "skos:closeMatch" is a sub-property of "skos:mappingRelation"
  - ○ Symmetry "skos:closeMatch" is an instance of "owl:SymmetricProperty"; for bidirectional mappings between "" and "" that are "rdf:type skos:Concept":
    - ▪ "skos:closeMatch"
    - ▪ "skos:closeMatch"
- **exact match**: skos:exactMatch
  - ○ Definition (from SKOS-REFERENCE): the property "skos:exactMatch" is used to link two concepts, indicating a high degree of confidence that the concepts can be used interchangeably across a wide range of information retrieval applications. The "skos:exactMatch" is a sub-property of "skos:closeMatch" and, in addition, it is an instance of "owl:TransitiveProperty". Also, "skos:exactMatch" is disjoint with each of the properties "skos:relatedMatch", "skos:broadMatch" and "skos:narrowMatch".
  - ○ Symmetry "skos:exactMatch" is an instance of "owl:SymmetricProperty"; for bidirectional mappings between "" and "" that are "rdf:type skos:Concept":
    - ▪ "skos:exactMatch"
    - ▪ "skos:exactMatch".

2.6 Recommender

The Recommender tool allows you to get recommendations for the most relevant semantic resources based on an excerpt from an ecological text or a list of keywords (see Figure 17). The semantic resource ranking algorithm used by the Recommender evaluates the appropriateness of each semantic resource to the input using a combination of four evaluation criteria:

- Coverage: to what extent does the semantic resource represent the input? The Recommender invokes the Annotator service to obtain all the annotations for the input and then uses those annotations to compute a coverage score for each semantic resource

- Acceptance: how well-known and trusted by the ecological community is the semantic resource? The number of visits to the semantic resource page in EcoPortal and the presence or absence of the semantic resource in UMLS are used to compute an acceptance score for each semantic resource

- Detail of knowledge: what is the level of detail provided by the semantic resource for the input data? It is computed using the number of definitions, synonyms and properties of the semantic resource classes that cover the input data

- Specialisation: how specialised is the semantic resource to the input data's domain? This is calculated using the number and type of the annotations done with the semantic resource and the position of each annotated class in the semantic resource hierarchy. The result is normalised by the size of the semantic resource, in order to identify small semantic resources that are specialised to the input data.
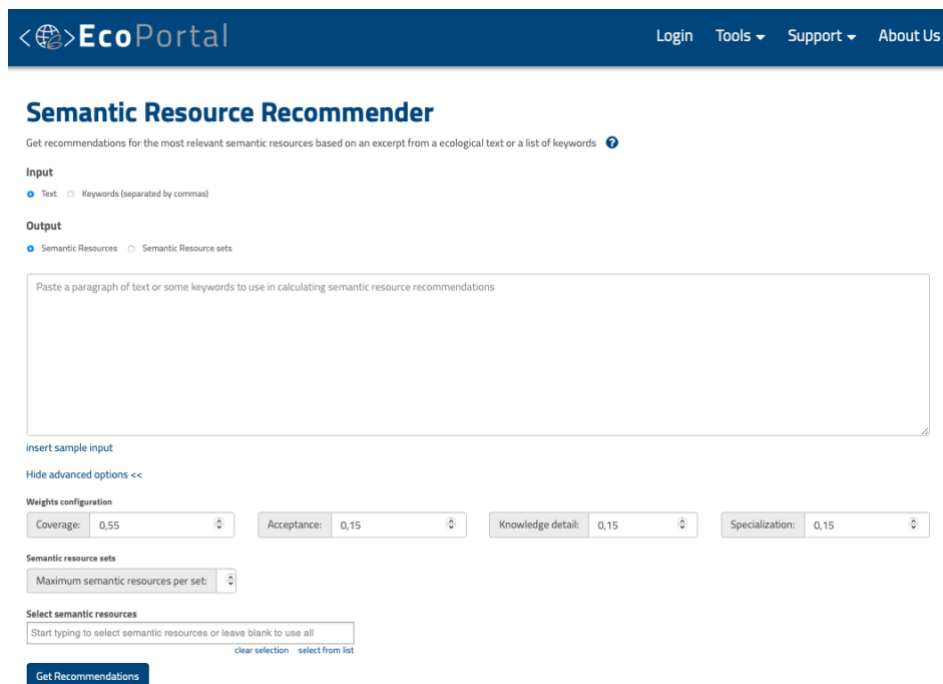
Figure 17: The recommender tool

For each of these four criteria, a score is computed. Then, the scores obtained are weighted and aggregated into a final score for the semantic resource. The default weights used to aggregate the scores can be changed in the advanced options section.

Note that besides the output of the semantic resource, which returns a ranked list of semantic resources, the Recommender also has a semantic resource sets output, which allows you to find those semantic resource sets that improve the coverage provided by individual semantic resources.

To retrieve semantic resource recommendations programmatically, please see our Ontology Recommender REST API documentation.

For more details about the workings of the Recommender and to cite it, please visit Martínez-Romero, M., Jonquet, C., O'Connor, M. J., Graybeal, J., Pazos, A., Musen, M. A. (2017). NCBO Ontology Recommender 2.0: An Enhanced Approach For Biomedical Ontology Recommendation. Journal of Biomedical Semantics, 8(21), 1-22

2.7 VocBench

The VocBench tool allows you to manage OWL ontologies, SKOS (/[XL]) thesauri, Ontolex-lemon lexicons and generic RDF datasets. It is a web-based, multilingual, collaborative development platform, designed to meet the needs of semantic web and linked data environments. VocBench

business and data access layers are realized by Semantic Turkey, an open-source platform for Knowledge Acquisition and Management realised by the ART Research Group at the University of Rome Tor Vergata]. EcoPortal uses VocBench as an editor for its resources with a customised version (v.8.0.2-beta.0).

The official documentation of VocBench can be consulted at the following link. For a detailed description of the VocBench version for EcoPortal, please refer to the following page.

Note: At this moment it is not possible to make a single registration on EcoPortal and VocBench. For now, you have to register separately for both services.

3. Publishing Semantic Resources

3.1 How to publish?

In order to publish a semantic resource on the EcoPortal website, you first need to Create an Account. Once your account is set up, you can Login to start the publishing process.

From the main EcoPortal page, navigate to Tools -> Publish. Alternatively, if you are on the Browsing page (from Tools -> Browse), you may publish your semantic resource by clicking on the Publish New Semantic Resource button.

In the resulting form, you will be asked to fill out some metadata about your semantic resource as shown in Figure 18:

- Name: name of the semantic resource
- Acronym: semantic resource acronym
- Administrators: you can select the administrators of the resource. Please note that they must be registered in EcoPortal otherwise they are not shown on the users list. The administrators of the resource can edit the list and make submissions
- Viewing restriction: you can select whether the resource is to be submitted for public viewing (i.e., all users, even ones not registered on the EcoPortal, may see it on the resources list and may display its contents) or for private viewing only, indicating in the form the accounts that can access it
- Categories: you can select the categories to which the resource belongs, for example to improve the search tools. For the moment, this functionality is not active. It will be activated in the near future, with the incorporation of categories from a published thesaurus
- View: you can check "this semantic resource is a view of" if you want to create a new semantic resource as a view of a selected semantic resource from the dropdown field.

You can also check the "Subscribe to email notifications for new notes" box if you are interested in receiving notifications when users of the EcoPortal community post notes on your published resource.

Figure 18: Form to publish a new sematic resource

Once you have filled out the form, click the "Create semantic resource" button in order to create the summary page for your semantic resource. Finally, on the summary page, clicking on the "Add submission" link you can fill in the new submission form (see Figure 19) where you will be able to specify:

- Format: here you can specify whether you have used the Unified Medical Language System (UMLS), the Open Biological and Biomedical Ontology (OBO), or the Simple Knowledge Organisation System (SKOS) to create your semantic resource. Alternatively, if you're uploading an OWL ontology, you may choose OWL from the dropdown menu; *in this case, and if your annotation properties have meaningful names (e.g., dc:description), please refrain from changing the specified default properties (Preferred Name Property, Synonym Property, Definition Property, and Author Property); map the default properties to your annotation properties only if the latter have coded names (e.g., BFO_0000179)*

- Status: submission release status (Alpha, Beta, Production, Retired)

- Release Date: date of issue of the submission

- Location: you can choose whether to upload only metadata for your resource or metadata, class/concept and properties. For this latter choice, you can upload the resource using the URL or selecting a file on your local file system

- Home Page: URL of the home page of the resource if it exists

- Documentation Page: URL of the resource documentation if it exists

- Publications Page: URL of articles or other products (e.g., book, book chapter, report) in which the semantic resource is referenced (e.g., DOI of a scientific article).

Figure 19: Form to add a new submission for a semantic resource

Once you have created the semantic resource submission, usually right away you can access its basic information and possibly its classes, if there are not too many of them. More detailed indexing takes place asynchronously, often scheduled at night. All of that should be finished by the next day (Pacific Time); if not, please send an email to the support list. Once the semantic resource has been indexed, you can fill in the submission information by specifying the EcoPortal metadata.

EcoPortal Metadata Schema

EcoPortal Metadata Schema is based on the metadata schema provided by BioPortal, and has been extended with some metadata fields from DataCite Metadata Schema v4.3. In particular, for this latter, we integrated only the metadata needed to obtain the DOI (Digital Object Identifier), which are: identifier, creator, title, publisher, publication year and resource type. In the table below you can see all the metadata fields used in EcoPortal.

ID			Field	Description	Cardinality
1			Identifier	The Identifier is a unique string that identifies a resource.	1
	1.1		identifier type	The type of Identifier.	1
2			Creator	The main researchers/institution/organization involved in producing the resource, in priority order.	1..∞
	2.1		creatorName	The full name of the creator.	1
		2.1.1	nameType	The type of name.	0..1
	2.2		givenName	The personal or first name of the creator.	0..1
	2.3		familyName	The surname or last name of the creator.	0..1
	2.4		nameIdentifier	It uniquely identifies an individual or legal entity, according to various schemes.	0..∞
		2.4.1	nameIdentifierScheme	The name of the name identifier scheme.	1
		2.4.2	schemeURI	The URI of the name identifier scheme.	0..1
	2.5		affiliation	The organisational or institutional affiliation of the contributor.	0..∞
		2.5.1	affiliationIdentifier	It uniquely identifies the organizational affiliation of the creator.	0..1
		2.5.2	affiliationIdentifierScheme	The name of the affiliation identifier schema.	1
		2.5.3	SchemeURI	The URI of the affiliation identifier schema.	1
3			Title	The name or title by which a resource is known. May be the title of a dataset or the name of a piece of software.	1..∞
	3.1		titleType	The type of title.	0..1
4			Publisher	The name of the entity that holds, archives, publishes prints, distributes, releases, issues,	1

				or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role.	
5			PublicationYear	The year when the data was or will be made publicly available. In the case of resources such as software or dynamic data where there may be multiple releases in one year, include the Date/dateType/dateInformation property and sub-properties to provide more information about the publication or release date details.	1
10			ResourceType	The description of the resource.	1
	10.1		resourceTypeGeneral	The general type of a resource.	1

Note: The fourth column, Cardinality, indicates the quantity constraints for the metadata field. The possible values are:
- 0-∞ = optional and repeatable
- 0-1 = optional, but not repeatable
- 1-∞ = mandatory and repeatable
- 1 = mandatory, but not repeatable.

In this metadata form, you can also ask for a DOI for the semantic resource that you are publishing. As you can see in Figure 20, there are different options:
- if you do not have the DOI for that semantic resource, you can request it by selecting "None" in the Type field and selecting the checkbox "I want to request a DOI"
- if you already have the DOI for that semantic resource, select "DOI" in the Type field and enter the DOI in the Identifier field
- if you have another type of identifier, select "Other" in the Type field and enter the ID in the Identifier field.
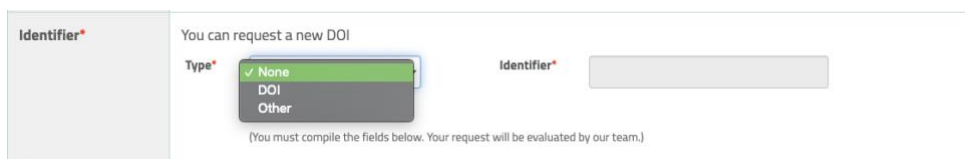
Figure 20: How to ask for a DOI when publishing a semantic resource

3.2 How to update the semantic resource?
If you are logged in as a manager of the semantic resource and you access the summary page in the menu area, the following sections will appear:
- Edit settings

- Add submission
- Edit submission.

Edit settings

This section allows you to change the settings for the semantic resource (e.g., name, administrators, viewing restrictions, categories, etc.). Note that you cannot change the semantic resource acronym, because EcoPortal uses this as a primary key and identifier. If you want to have another acronym, you have to submit a new semantic resource, using the new acronym, and then ask us to delete or deprecate the previous semantic resource.

Add submission

This section allows you to update the previously submitted semantic resource with a new version and to update the metadata for the submission. You can update the semantic resource in one of the following two ways:

- if the original submission was set to automatically update from a source URL, EcoPortal examines the content at this URL nightly, and re-submits the content if it has changed in any way (determined by the checksum). If the updated submission does not parse, the semantic resource manager(s) will be notified via email
- if the original submission was not set to be automatically updated, or if you want to immediately update the semantic resource, you can upload the semantic resource directly by selecting a file from your computer. This will begin the upload process, and schedule the semantic resource to be parsed shortly thereafter. (After successful submission, don't forget to re-enter the URL and settings for the automatic update process, if you want that to be enabled.) Usually, semantic resource parsing is completed quickly, from a few minutes to an hour, except for large semantic resources; semantic resource indexing and automatic mappings occur overnight (Pacific Time).

Edit submission

This section allows you to update the metadata for the latest submission of your semantic resource, by editing the information on the metadata form. Moreover, you can also replace the URL or the local file if you made a mistake when you have submitted. Once you have edited the form, click on the "Save Submission" button. If you do not change the "Location" settings of the resource, only the changes to the metadata will be saved.

3.3 Creating and editing semantic resources in VocBench

VocBench EcoPortal is an active service on the EcoPortal platform that allows you to edit and publish specific semantic resources in the ecological domain. To create a new project in VocBench EcoPortal you have two possibilities:

a) If you are not registered you have to do it at the following link (see Figure 21). After filling in some personal data, and if you already have a semantic resource published in EcoPortal, insert into "Existing semantic resource?" the full name and acronym, e.g., Fish Traits Thesaurus (FISHTRAITS). By default, the last version on EcoPortal is loaded, otherwise you have to specify the version from which to start editing in VocBench. NOTE: to request editing of an EcoPortal semantic resource, you must be the owner of that resource.

b) Otherwise, in order to create a new semantic resource, you have to specify:
  - **New semantic resource:** the name of the project, e.g., Fish Traits Thesaurus. NOTE: it is suggested to specify the full name that will be used in EcoPortal
  - **BaseURI:** the URL necessary for the creation of a project
  - **Model:** the model of the new project which can be OWL or SKOS. NOTE: To understand the support of the SKOS model in EcoPortal we recommend reading the following section

c) If you are already registered, click on the "Request Project" button in the Home Page and fill in the form as shown in Figure 22:

Figure 21: Registration in VocBench and creation of a new semantic resource

Figure 22: Request project in VocBench

After this, the administrator will create a new project assigning you the role of Project Manager. The administrator will specify all technical details needed for the creation of the new project, e.g., lexicalisation, history, validation and blacklisting and the type of the repository access, which is a remote triple store like GraphDB store.

Then in the list of projects, there should be a new item for the (empty) project as shown in Figure 23. The open folder indicates that the project is open, but it may be necessary to click on the radio button on the left to access the project.

Figure 23: List of projects in VocBench

Subsequent steps require moving to the "Data" tab to edit and then to export the data. In the "Scheme" tab (see Figure 24), click on the create button to create a new skos:ConceptScheme ([]*):
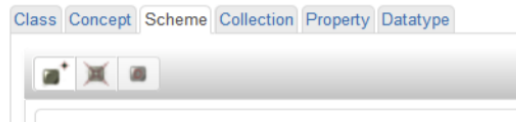
Figure 24: Scheme tab of the project in VocBench

In the creation dialog (see Figure 25), specify a label for the concept scheme being created e.g., "continents" in English and then click on the Ok button.
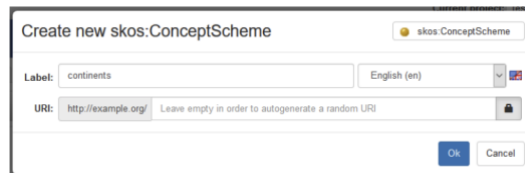
Figure 25: Creation of a concept scheme in VocBench

Then, select the newly created scheme as in Figure 26.
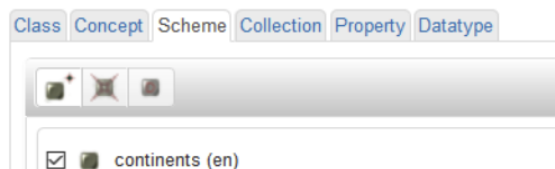
Figure 26: Selection of a newly created scheme in VocBench

In the Concept tab (see Figure 27), click on the create button to create a new skos:Concept (()*).
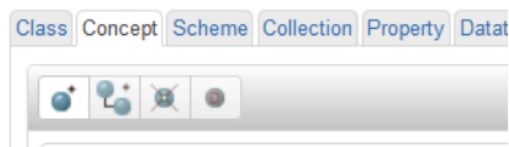
Figure 27: Concept tab of the project in VocBench

In the creation dialog (see Figure 28), specify a label for the concept being created e.g., "Africa" in English and then click on the "Ok" button.
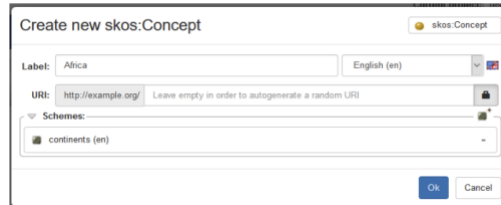
Figure 28: Creation of a new concept in VocBench

Do the same to create other concepts:
- America
- Asia
- Europe
- Oceania

At the end, the concept tree should contain a flat list of the five continents (see Figure 29).
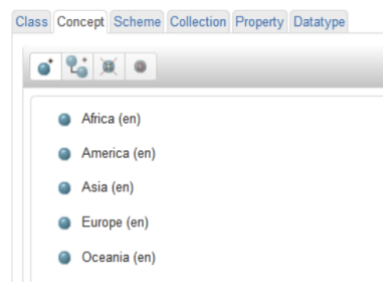
Figure 29: The concept tree in VocBench

To understand in detail the different aspects of the user experience on VocBench, we recommend reading the following link.

4. EcoPortal Deployer for VocBench 3

The EcoPortal Deployer for VocBench 3 allows the publication of semantic resources (specifically OWL ontologies or SKOS thesauri) on EcoPortal. The EcoPortal Deployer can be used in an export configuration whose target is a triple store: the EcoPortal Deployer directly consumes the RDF data being exported, without requiring that they be serialised beforehand into a byte sequence conforming to some format. Indeed, the serialisation occurs internally as part of the implementation of the submission protocol. The deployer provides a dedicated configuration for EcoPortal and its new mandatory metadata based on the Data Cite Metadata Schema. In addition, it is possible to configure the deployer also for generic OntoPortal catalogues (e.g., BioPortal).

4.1 How to use

You, as project manager, can use the Deployer by selecting the Export Data item under the "Global Data Management" menu (in the top right corner of the user interface of VocBench 3 as shown
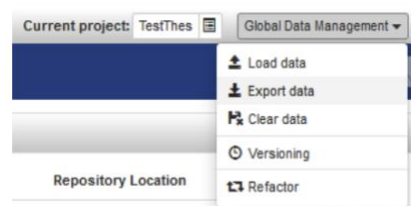
Figure 30: Global Data Management menu in VocBench

in Figure 30).

The "Export data" panel makes it possible to define complex export configurations. The panel consists of three sections as shown in Figure 31:

- Graphs to export: this section allows you to select which graphs will be exported. The default selection just contains the graph of your project, named with the base URI, which contains the data being edited
- Export transformations: this section allows you to configure a sequence of RDF Transformers, which can be used to manipulate the data being exported (e.g., filter out triples, add triples, replace triples, etc.)
- Deployment: this section allows you to configure the destination of the data being exported and, optionally, transformed. You can choose from the dropdown menu whether the data will be saved to a file, exported to a triple store, or sent to a custom destination. As already discussed, the target of the EcoPortal Deployer is considered a triple store.
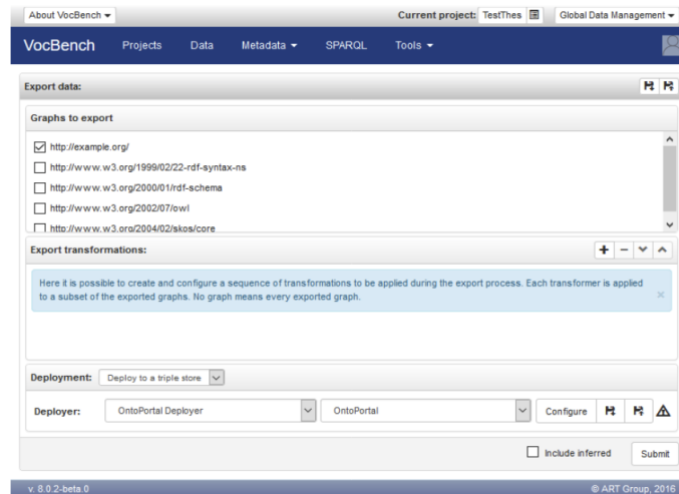
Figure 31: The "Export data" panel in VocBench

After selecting "Deploy to a triple store" in the Deployment dropdown, the Deployer dropdown list will appear. Here you have to select "OntoPortal Deployer" and the item labelled EcoPortal to enable the specific characteristics of the deployer (e.g., additional configuration parameters) dedicated to EcoPortal, as depicted in Figure 32.
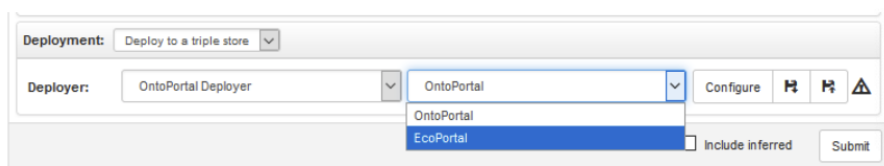
Figure 32: Selecting the appropriate deployer in VocBench

The warning sign on the right of the widget associated with the EcoPortal Deployer indicates that it requires further configuration. A click on the "Configure" button will open the dialog shown in Figure 33 to edit the chosen configuration. Here below we summarise the configuration for EcoPortal, which consists of several fields:

- API Base URL: the base URL of the EcoPortal REST API. If this parameter is omitted, then the deployer defaults to the official base URL of EcoPortal: http://ecoportal.lifewatch.eu:8080/

- API key (mandatory): the API key that will be used to authorize the semantic resource submission. Your user API key can be found in the user's account page

- Acronym (mandatory): the acronym that identifies the semantic resource for which the submission is being made

- Description: the textual description of the semantic resource

- Version: the version of the semantic resource

- Format (mandatory): the format of the semantic resource (either OWL or SKOS)

- Status (mandatory): the status of the semantic resource (either alpha, beta, production, or retired)

- Release Date (mandatory): the release date of the semantic resource, formatted as yyyy-mm-dd

- Contacts (mandatory): at least one contact for the semantic resource. Additional contacts can be added by clicking on the plus button on the right. Each contact should be formatted as name (email), for example: Mario Bianchi (mario.bianchi@example.org)

- Homepage: the address of the main web page of the semantic resource

- Documentation: the address of a web page providing documentation for the semantic resource

- Publications (mandatory): the address of a web page listing publications for the semantic resource

- Creators (mandatory): the main researchers/institution/organization involved in producing the resource, in priority order

- Titles (mandatory): at least one name or title by which the resource is known

- Publisher (mandatory): the name of the entity that holds, archives, publishes, prints, distributes, releases, issues, or produces the resource

- Publication year (mandatory): the year when the resource was or will be made publicly available

- Resource type (mandatory): the description of the resource. The format is open, but the preferred format is a single term of some detail

- Resource type general (mandatory): the general type of resource.

Figure 33: EcoPortal configuration in VocBench

The fields marked with an asterisk (*) are mandatory and all other fields are optional. Once the deployer has been configured, it is possible to click on the "Submit" button on the bottom of the "Export Data" panel to create the submission to EcoPortal. After a little while you should receive a confirmation message as shown in Figure 34.
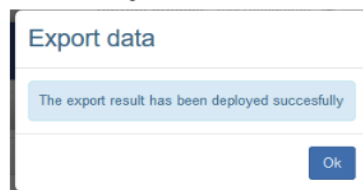
Figure 34: Confirmation message for correctly exporting data in VocBench

4.2 Data transformation

EcoPortal imposes some constraints for the semantic resources in SKOS format (see the documentation) and it requires (among other requirements):

- the use of the property skos:hasTopConcept

- the use of plain (no- reified) SKOS labels.

The first requirement is not met by VocBench by default (since it uses the inverse property skos:isTopConceptOf). However, an RDFTransformer can be used to add the triple expressing the relationship in the opposite direction. On the section "Export transformations", click on the plus button and then select the "SPARQL RDF Transformer" as shown in Figure 35.
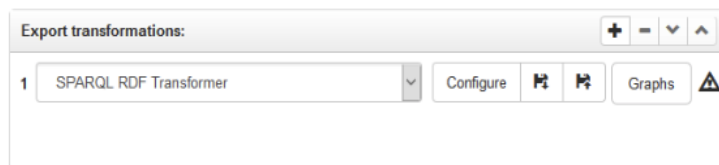
Figure 35: SPARQL RDF Transformer as export transformation in VocBench

Click on the "Configure" button and specify the following filter in the configuration dialog (note that the "filter" field can be enlarged by dragging the grabber on the bottom right corner of the text area) as shown in Figure 36:

```
INSERT {
        ?s <http://www.w3.org/2004/02/skos/core#hasTopConcept> ?c
        }WHERE {
        ?c <http://www.w3.org/2004/02/skos/core#topConceptOf> ?s
        }
```
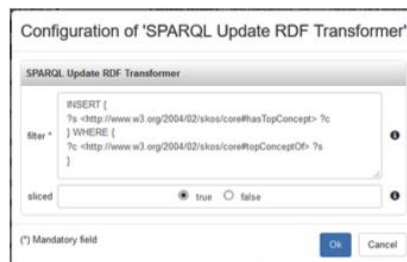

Figure 36: Configuration of the SPARQL Update RDF Transformer in VocBench

The second requirement can be met by using SKOS as a lexicalisation model. If a project uses SKOS-XL, it can be used as a dedicated RDFTrasformer to generate the plain labels from the reified ones.

4.3 Versioning

Versioning is another item in the Global Data Management menu that allows you to save a snapshot of the data being edited.

Our suggestion is not to export the "CURRENT" version that denotes the continually evolving data

Figure 37: Versioning function in VocBench

being edited in a project. Indeed, any submission should be done through the following procedure:

- create a dump of the data and assign it a version identifier
- temporarily switch to the dumped data
- make the submission using the already described export procedure: the field "version" in the configuration of the EcoPortal Deployer should match the version identifier of the data dump.

4.4 Verification

Login into EcoPortal and open the page of your target semantic resource.

The semantic resource submission may not immediately appear on the EcoPortal web application in the list of the semantic resource submissions. Submissions are handled asynchronously: at first, we might see that the submission has just been Uploaded, but subsequently we should see that it is marked with Parsed, Indexed, Metrics and Annotator. In some circumstances, the new submission may not be visible at all just inside the web page. In this case, we can verify that EcoPortal has received the submission by retrieving the submissions using the REST API.

4.5 SKOS Support

Support for SKOS vocabularies in EcoPortal

EcoPortal is a web-based portal for accessing and sharing semantic resources. The application accepts semantic resource submissions in OWL and OBO format, and SKOS vocabularies that contain particular constructs. This wiki page documents the minimum set of SKOS constructs that must be present in a SKOS vocabulary for EcoPortal to accept and handle the submission properly. Please note that the SKOS constructs described here are handled only for vocabularies that are identified as SKOS when they are submitted to EcoPortal.

Required SKOS constructs

- skos:Concept: Concepts are the fundamental elements of SKOS vocabularies and are asserted using the skos:Concept class, e.g.: <http://www.example.com/animals> rdf:type skos:Concept.

In SKOS vocabularies, EcoPortal only treats the SKOS concept assertions as concepts to be displayed. If the vocabulary contains other assertions about other types of concepts, EcoPortal

will not treat these as concepts in any of its displays or features.

See the W3C's SKOS System Primer and SKOS Reference for concept documentation and examples:
- https://www.w3.org/TR/2009/NOTE-skos-primer-20090818/#secconcept
- https://www.w3.org/TR/skos-reference/#concepts.

Note: Some OWL ontologies declare the SKOS namespace to facilitate minimal use of SKOS constructs for things like labels (e.g., skos:prefLabel, skos:altLabel) or mappings (e.g., skos:exactMatch, skos:broaderMatch). In these cases, the proper format for new ontology submissions is OWL, not SKOS.
- skos:ConceptScheme & skos:hasTopConcept: For every semantic resource entry in EcoPortal, the application provides a tabbed interface with various views of the semantic resource data, e.g., a "Classes" tab with a tree structure to graphically depict the hierarchical collection of semantic resource classes.

In the case of SKOS vocabularies, EcoPortal determines which concepts to display as roots in the concept tree by querying vocabulary content for occurrences of skos:hasTopConcept property assertions. Top concepts are the most general concepts contained in SKOS concept schemes (an aggregation of one or more SKOS concepts).
The following example, taken from the SKOS System Primer, shows how to define a concept scheme and link it to the most general concepts it contains:

```
@prefix skos: <http://www.w3.org/2004/02/skos/core#>
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix ex: <http://www.example.com/>
ex:animalThesaurus rdf:type skos:ConceptScheme
skos:hasTopConcept ex:mammals
skos:hasTopConcept ex:fish.
```

SKOS vocabularies submitted to EcoPortal must contain a minimum of one concept scheme and top concept assertion. See the SKOS System Primer and SKOS Reference for more documentation of concept schemes and top concepts:
- https://www.w3.org/TR/2009/NOTE-skos-primer-20090818/#secscheme
- https://www.w3.org/TR/skos-reference/#schemes.

If your vocabulary declares more than one concept scheme, all of the top concepts will be aggregated and displayed as root level concepts. The EcoPortal user interface does not provide support for grouping top level concepts by concept scheme. It is highly recommended to declare an owl:Ontology, especially for metadata annotations.

Hierarchy in SKOS vocabularies

The only semantic relationship in SKOS vocabularies that EcoPortal uses to construct and display concept hierarchies is the skos:broader property.

```
ex:mammals rdf:type skos:Concept;
skos:inScheme :myScheme;
skos:narrower ex:cat;
skos:prefLabel "mammals"@en;
skos:broader ex:animals.
```

Other properties used to denote hierarchical relationships like skos:narrower, skos:broaderTransitive, and skos:narrowerTranstive, are ignored.

Metrics data for SKOS vocabularies

EcoPortal uses the OWL API for parsing all ontology and vocabulary submissions, as well as for the calculation of metrics data. The OWL API treats SKOS vocabularies as RDF files containing classes and instances. According to the SKOS Reference, concepts are instances of owl:Class, and thus are counted as instances (a.k.a. "individuals").

When viewing metrics tables in the EcoPortal user interface, the value for the "NUMBER OF INDIVIDUALS" corresponds to the number of concepts in any given SKOS vocabulary.

SKOS-XL

Currently EcoPortal offers no support for the SKOS eXtension for Labels (SKOS-XL). A suggested workaround for SKOS vocabularies that make use of SKOS-XL, is to dump the value of labels (i.e., skosxl:literalForm of skosxl:*Label instances) into the corresponding skos:*Label property.

Example of valid SKOS

This example provides a simple illustration of the composition of a SKOS file that complies with the above constraints.

Example header

The header shown here defines a few typical namespaces that may be useful. The last namespace is the one that defines this SKOS vocabulary. Ideally, the IRI defining the myskosid namespace is the resolvable location of the SKOS ontology.

```
<xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:skos="http://www.w3.org/2004/02/skos/core#"
  xmlns:dct="http://purl.org/dc/terms/"
  xmlns:myskosid="https://example.org/ontologies/myskosontology/"
>
```

Example semantic resource description

In the rdf:type item, this namespace is declared as the ConceptScheme. The ConceptScheme does not have to be the same as the namespace of the ontology.

Other metadata are provided as examples of good practices in ontology metadata. The dct:creator does not have to be an ORCID ID, but a unique identifier is an ideal way of naming a creator (whether individual or organization).

This semantic resource has only 2 concepts (to be defined below), hence only 2 skos:hasTopConcept declarations.

```
<rdf:Description rdf:about="https://example.org/ontologies/myskosontology/">
   <rdfs:label xml:lang="en">Example SKOS ontology for EcoPortal</rdfs:label>
   <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#ConceptScheme"/>
   <rdfs:comment xml:lang="en">Example created to simplify understanding and creation of
a SKOS vocab for EcoPortal</rdfs:comment>
   <dct:created rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2020-09-
16</dct:created>
   <dct:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2020-09-
16</dct:modified>
   <dct:license rdf:resource="https://creativecommons.org/licenses/by/4.0/"/>
   <dct:creator rdf:resource="https://orcid.org/0000-0001-6875-5360"/>
   <skos:hasTopConcept
rdf:resource="https://example.org/ontologies/myskosontology/fragmentid001"/>
   <skos:hasTopConcept
rdf:resource="https://example.org/ontologies/myskosontology/fragmentid002"/>
 </rdf:Description>
```

Example term definitions

This section shows the two concepts and a few typical annotations about those concepts. The first rdf:Description line of each group names the concept that is being defined in the indented lines that follows.

The rdf:Type and skos:prefLabel are required annotation content for EcoPortal to work effectively. Other items are optional.

The skos:topConceptOf is not strictly required for EcoPortal SKOS semantic resources, but provides useful contextualiation if there is more than one topConcept.

```
    <rdf:Description
  rdf:about="https://example.org/ontologies/myskosontology/fragmentid001"">
      <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
      <skos:prefLabel xml:lang="en">First concept</skos:prefLabel>
      <skos:definition xml:lang="en">The very first example provided as part of this
  semantic resource.</skos:definition>
      <skos:topConceptOf rdf:resource="https://example.org/ontologies/myskosontology/"/>
    </rdf:Description>
    <rdf:Description
  rdf:about="https://example.org/ontologies/myskosontology/fragmentid002"">
      <rdf:type rdf:resource="http://www.w3.org/2004/02/skos/core#Concept"/>
```

```
        <skos:prefLabel xml:lang="en">Second concept</skos:prefLabel>
        <skos:definition xml:lang="en">The very first example provided as part of this
    semantic resource.</skos:definition>
        <skos:topConceptOf rdf:resource="https://example.org/ontologies/myskosontology/"/>
    </rdf:Description>
```

Closing XML
Needed for a complete, parseable RDF file!   `</rdf:RDF>`
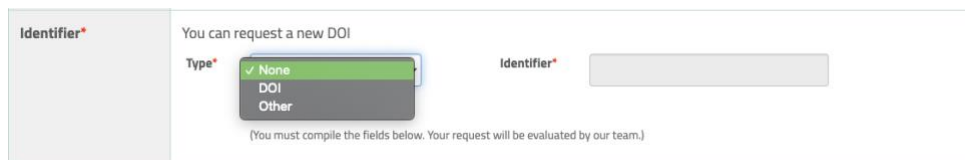
5. DOI Services

How to get a DOI for a Semantic Resource?
LifeWatch ERIC is a direct member of [DataCite](#) and can provide a Digital Object Identifier (DOI) for semantic resources that do not already have one, taking advantage of the EcoPortal – DataCite connection. This makes it possible to cite the semantic resources in scientific publications or elsewhere in a reliable and sustainable way. EcoPortal assigns a DOI only upon verification and validation of the semantic resource. This implies that the DOI provision can be refused if the resource does not match with its quality criteria or its domain (that is ecology).

The DOI is connected to a metadata description of the resource as well as to a digital location, such as a URL, where all the details about the resource are accessible.
As already reported in detail in the [EcoPortal Metadata Schema](#) section, you can request a DOI when specifying the Identifier field of the metadata form by selecting "None" in the Type field and the checkbox "I want to request a DOI" (see Figure 38).

Figure 38: How to request a DOI

6. Developer

6.1 Programming with the EcoPortal APIs

All the content that the EcoPortal uses (and more!) is available via REST services by means you can:

- access all EcoPortal semantic resources, their different versions, and metadata for those versions
- access information about any semantic resource concept in EcoPortal (with definitions, synonyms, and other properties)
- search across all semantic resources in EcoPortal
- get hierarchy information for EcoPortal semantic resources (such as parents, children, or siblings of a class, roots or leaves of a class hierarchy).

The EcoPortal REST API Documentation is available here: http://ecoportal.lifewatch.eu:8080/documentation.

6.2 Getting an API key

Use of the EcoPortal REST APIs requires an API key. To retrieve your API key, you have to be logged in the EcoPortal website where the key will be shown in plain text in your account page (see Figure 39). If you don't have an account, you need to create one

6.3 Using EcoPortal technology in your project
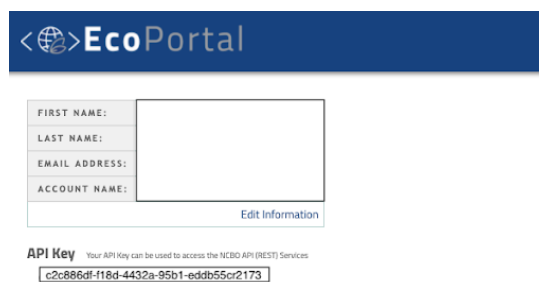
Figure 39: How to retrieve your API key

There are several ways in which you can use the EcoPortal technology for your own website or in your application. Here some use cases:

- I have a list of terms that were used to annotate a dataset, how can I see if any of these terms exist in standard semantic resources? Use the "Search" web service to search terms in all EcoPortal semantic resources. If you have a limited number of terms, searching each one individually via the EcoPortal Search tool with the user interface may be sufficient. Otherwise, for a long list of terms, use the "Search" web service directly.

- I would like to find all the terms mapped from my semantic resource to other semantic

resources in EcoPortal. Use the "Mapping" web service to find terms in other EcoPortal semantic resources that are similar to the terms in your semantic resource. These mappings include those generated manually (DbXrefs) and programmatically (LOOM).

▪ I have a web form including text fields which I would like to populate with semantic resource terms, e.g., to select relevant terms from a specific EcoPortal semantic resource. Use the "Autocomplete" widget. Click on the "Get Code" button for the code and instructions on how to add this JavaScript to your web form.

▪ I would like to add the graph displaying my semantic resource on my website. Use the "Visualization" widget. Click on the "Get Code" button for the code and instructions on how to add this JavaScript to your website.

▪ I would like to get all terms from a semantic resource and re-format the data to use in my own system. Use the "Get All Terms" web service to get all the terms and properties for the semantic resource that you are interested in.

▪ I need to navigate through a semantic resource and go through the semantic resource hierarchy for my software application. Is there a web service to do this? Use the "Get All terms" web service.

▪ How can I limit the selection of semantic resources used with the NCBO Annotator to a given Category or Domain? Use the "List all semantic resources" and "Get Groups/Domains" web service to identify the group/domain of interest.

EcoPortal widgets

EcoPortal widgets are snippets of HTML or JavaScript code that you can embed in your website/web form to use the EcoPortal functionalities. Using EcoPortal widgets is just one of the ways in which you can use EcoPortal technologies on your website/application.

How to use EcoPortal widgets

You can add any EcoPortal widget to your HTML page or web form by following these steps:
▪ find the semantic resource that you are interested in by browsing the list of EcoPortal semantic resources (e.g., Fish Traits Thesaurus)
▪ click on the semantic resource name to get to the page with Details for that semantic resource (e.g., the details for Fish Traits Thesaurus)
▪ in the semantic resource details page, go to the "Widgets" tab
▪ select the widget that you want to use
▪ click the button below the widget to get usage instructions.

Types of EcoPortal widgets and use cases

- **Term-selection field on a form**: you can add a text field to your web form that will allow users to enter a term from a controlled vocabulary (e.g., terms from a single semantic resource). **Example use case:** let us suppose that you are running a tissue microarray database and that users can upload sample descriptions to your database by means of a web form. Usually, there is a field to specify the diagnosis related to the tissue sample that the user is filling in. This field is often a text-box or a dropdown menu, populated with a list of controlled terms. The free text-box is prone to errors, while the dropdown becomes unmanageable with too many terminologies. The EcoPortal term-selection widget allows users to easily select a term from an ontology or a controlled vocabulary to fill in the specific field. For example, when the user starts typing "fish", the term "fish traits'' pops up.
  What does the term-selection field get you?
  - Auto-complete the text field so that you don't need to type the whole term
  - The controlled vocabulary provides consistency in the manner in which different users use a specific term
  - When a new version of your controlled vocabulary becomes available in EcoPortal, the widget will automatically use that new version.

- **Semantic resource search widget**: you can add a search box to your website that will allow users to search a specific EcoPortal semantic resource. When the user selects the term of interest (with the help of the auto-complete feature), she/he can move directly to the EcoPortal summary page of the corresponding concept.

- **Semantic resource visualization widget**: you can put a widget on your website that shows your semantic resource of interest, or a part of it, and obtain the same result as in the "Visualize" tab in EcoPortal.

- **EcoPortal Ontology Tree Widget**: you can put a widget on your website that displays the semantic resource tree, or a part of it, with the same result obtained in the "Visualize" tab in EcoPortal.

Term-selection field on a form

To add a field to your web form that allows users to retrieve a term from the semantic resource of interest, do the following:

- download the form_complete.js file and put it on your server

- in the header of the page where you want the field, include the form_complete.js file

- in the form, specify the field class in the following format: bp_form_complete-{semantic_resource_acronym_list}-{value}
  Example: bp_form_complete-FISHTRAITS-uri will use FISH TRAITS Thesaurus (the acronym is FISHTRAITS) and will put the term URI in the field after the user selects the term from the pull-down list
  Note: In addition to single ontology IDs, you can use a list: bp_form_complete-FISHTRAITS, ENVTHES-uri OR you can use 'all' to search across all EcoPortal semantic resource: bp_form_complete-all-uri

- you can use the following parameters to select which value will be placed into the user-visible input field:

  - uri: the complete URI of the term (e.g., http://thesauri.lifewatchitaly.eu/FishTraits/?tema=5);

  - shortid: the short ID of the term, as used in EcoPortal (e.g., ?tema=5);

  - name: the preferred name of the term (e.g., "Fish Trait").

- **Auto-generated hidden form elements**

In addition to the input element you defined, there are four hidden form elements that are automatically created and set when users select a term from the list. For example, if you create a field with this code:

```
<input type="text" name="a" class="bp_form_complete-all-uri" size="100"/>
```

The 'name' attribute is used to create the four following fields (note how the 'a' from the name attribute is appended to the id attribute):

```
<input type="hidden" id="a_ecoportal_preferred_name">
<input type="hidden" id="a_ecoportal_concept_id">
<input type="hidden" id="a_ecoportal_ontology_id">
<input type="hidden" id="a_ecoportal_full_id">
```

- **Optional attributes**

  - Include term definitions. Use data-bp_include_definitions="true" to display definitions with the list of terms that are returned. For example, use this code to display search results for 'heart' terms, with definitions, in the FISHTRAITS Thesaurus:

    ```
    <input type="text" name="m" data-bp_include_definitions="true"
    ```

```
                class="bp_form_complete-FISHTRAITS-name"/>
```

- Limit search to terms below a root term. Use data-bp_search_branch="{class_id}", replacing {class_id} with the class ID for the term to use as the branch root. For example, use this code to limit the search for 'FISH' to terms within the 'Fish Trait' branch of the FISHTRAITS Thesaurus:

```
<input type="text" name="n" data-bp_search_branch="Fish Trait"
class="bp_form_complete-FISHTRAITS-name"/>
```

- Object types:

  o Search 'class' and 'property' objects. Use data-bp_objecttypes="class,property". For example, use this code to search for 'function' among 'class' and 'property' objects from the "Adverse Event Reporting Ontology":

  ```
  <input type="text" name="o" data-bp_objecttypes="class,property"
  class="bp_form_complete-AERO-uri"/>
  ```

  o Search only 'property' objects. Use data-bp_objecttypes="property". For example, use this code to search for 'function' among only 'property' objects from the "Adverse Event Reporting Ontology":

  ```
  <input type="text" name="p" data-bp_objecttypes="property"
  class="bp_form_complete-AERO-uri"/>
  ```

  o Search 'class' and 'individual' objects. Use data-bp_objecttypes="class,individual". For example, use this code to search for 'seizure' among 'class' and 'individual' objects from the "Adverse Event Reporting Ontology":

  ```
  <input type="text" name="q" data-bp_objecttypes="class,individual"
  class="bp_form_complete-AERO-uri"/>
  ```

  o Search only 'individual' objects. Use data-bp_objecttypes="individual". For example, use this code to search for 'seizure' only among 'individual' objects from the "Adverse Event Reporting Ontology":

  ```
  <input type="text" name="r" data-bp_objecttypes="individual"
  class="bp_form_complete-AERO-uri"/>
  ```

Semantic resource search widget

To add a search widget to your HTML page that allows users to search for a specific semantic resource:

www.lifewatch.eu

- download the quick_jump.js file and put it on your server;
- copy the code below and paste it to your HTML page

```
<div id="bp_quick_jump"></div>
 <script type="text/javascript">
   var BP_ontology_id = "FISHTRAITS";
 </div>
 <script src="quick_jump.js" type="text/javascript" charset="utf-8"></script>
```

Note: if you want to use Quick Jump across multiple semantic resources:

- enter a comma-separated list of semantic resource ids: var BP_ontology_id = "FISHTRAITS, ENVTHES", or

- you can set the variable to 'all' to search all semantic resources in EcoPortal: var BP_ontology_id = "all".

- update the copied code by changing the path to the quick_jump.js file that has to point to the server location where you put the HTML file. For example, if you put the quick_jump.js file in the same directory as your HTML file, this is the code that you have to use:

```
<script type="text/javascript">
  var BP_ontology_id = "FISHTRAITS";
</div>
<script src="quick_jump.js" type="text/javascript" charset="utf-8"></script>
```

- to limit the term search to a particular branch of a semantic resource, include the following JavaScript in your page (remember to replace the conceptid with the ID for the term that you want to use as the root of your branch):

```
var BP_search_branch = "{class_id}"
```

- to display definitions with the list of terms that are returned, include the following JavaScript in your page:

```
var BP_include_definitions = true.
```

Semantic resource visualization widget

To add a widget visualizing your semantic resource in your HTML page, copy the code from the Widget page of your semantic resource of interest. Here is an example:

```
<iframe
src="http://ecoportal.lifewatchitaly.eu/widgets/visualization/?ontology=FISHTRAITS&class=S
```

```
ELECT_CLASS&apikey=YOUR_API_KEY"frameborder="0">
</iframe>
```

EcoPortal semantic resource tree widget
▪ **Description**

The semantic resource tree widget is an embeddable Flex application that allows users to interact with a tree to browse semantic resources. The widget can be configured to display a list of all the EcoPortal semantic resources, allowing users to select which one to view, or can be set to view a particular semantic resource. Moreover, there are several JavaScript helper functions for the widget, allowing it to be used as a fully functional component in other applications.

▪ **Location**

To use the tree widget, you have to add the following files:
● http://ecoportal.lifewatchitaly.eu/widgets/jquery.ncbo.tree.css
● http://ecoportal.lifewatchitaly.eu/widgets/jquery.ncbo.tree-2.0.2.js

or the minified versions:
● http://ecoportal.lifewatchitaly.eu/widgets/minified/jquery.ncbo.tree.min.css
● http://ecoportal.lifewatchitaly.eu/widgets/minified/jquery.ncbo.tree-2.0.2.min.js.

▪ **Setup**
jQuery is required in order to use the tree widget. If jQuery is not still in your project, you have to include it as follows:
```
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
```

Then, you have to create a div element to store the tree:
```
<div id="tree"></div>
```

initialize the tree with the following JavaScript code:
```
var tree = $("#tree").NCBOTree({apikey: "YOUR_API_KEY",ontology: "FISHTRAITS"});
```
The div you created will contain a tree showing the roots of the FISHTRAITS semantic resource. Additional setup options include:
● **autoclose:** to close other branches of the tree when expanding a new one
● treeClass: the class to identify the tree
● **autocompleteClass:** the class to identify the autocomplete (search) input
● **width:** the width of the tree
● **startingClass:** to start with this class selected (and expanded to) in the tree

- **startingRoot:** to start with this class as the only root displayed in the tree. This allows you to display a sub-tree or branch of your choosing
- **ncboAPIURL:** to use a different NCBO REST instance (not normally used)
- **ncboUIURL:** to use a different NCBO UI instance (not normally used).

▪ **Callbacks and event triggers**

You can interact programmatically with the tree by using JavaScript and jQuery. Callbacks can be passed when setting up the tree:

```
var tree = $("#tree").NCBOTree({apikey: "YOUR_API_KEY",ontology: "FISHTRAITS",
afterSelect: function(event, classId, prefLabel, selectedNode){console.log(classId);}});
```

The available callbacks with their arguments are:
- **beforeExpand,** arg: expandedNode
- **afterExpand,** arg: expandedNode
- **afterExpandError,** arg: expandedNode
- **afterSelect,** arg: classId, prefLabel, selectedNode
- **afterJumpToClass,** arg: classId.

You can also listen to these callbacks as jQuery events:

```
tree.on("afterSelect", function(event, classId, prefLabel,
selectedNode){console.log(classId);}
```

The events with their arguments are:

- **beforeExpand, arg:** event,expandedNode
- **afterExpand, arg:** event,expandedNode
- **afterExpandError, arg:** event,expandedNode
- **afterSelect, arg:** event,classId, prefLabel, selectedNode
- **afterJumpToClass, arg:** event,classId**.**

▪ **Public methods**

There are a variety of methods to call on the object that is returned when calling NCBOTree(). They can be used to get information about the state of the tree or to programmatically interact with it.

```
var tree = $("#tree").data("NCBOTree");
```

OR

```
var tree = $("#tree")[0].NCBOTree;
```

OR

```
var tree = document.getElementById("tree").NCBOTree;
```

Once you have the tree instance, you can call the following methods:

- **tree.selectedClass():** returns an object that represents the currently-selected class. The object includes attributes for ID, prefLabel, and URL. URL is the REST location of the class, and performing an HTTP GET on that URL will provide a JSON representation of the class

- **tree.selectClass(classId):** activates the class in the tree with the matching URI (classId). This will only work if the class is already visible

- **tree.jumpToClass(classId):** replaces the current tree with a version that is expanded with a path from the root to the given class. This method is triggered when selecting a class from the search field. The provided class will be selected in the tree when the tree returns

- **tree.changeOntology(ontologyAcronym):** replaces the current tree with the roots from the given semantic resource.