Fennec Documentation

Release 1.0.0

Markus J. Ankenbrand, Sonja Hohlfeld, Frank Foerster, Alexander

May 12, 2021

Contents

Contents:

Introduction

Fennec is an acronym for "Functional Exploration of Natural Networks and Ecological Communities". It is a web platform that facilitates enrichment of (taxonomically classified) community tables with trait data. For this purpose trait information from various sources is stored in a database. Users can then upload their community tables, map the organisms to entries in the database and automatically retrieve all relevant traits. Fennec provides basic visualization of trait composition in the community by integrating Phinch. Enriched community tables can be exported for use in external analysis tools like PhyloSeq.

Also read (and if you use FENNEC, please cite) our Manuscript (doi:10.1111/2041-210X.13060).

User manual

2.1 Quick start

To learn the main features of Fennec from a user perspective navigate your webbrowser to the public instance at https://fennec.molecular.eco The first thing you see is the startpage

The boxes show the number of organisms and trait entries in the current database. You can explore the organisms and traits in the database by using the navigation on the left hand side. However, in order to analyze projects it is necessary to log in (in the top right corner). You can login using your GitHub account or register a free Fennec account.

After login navigate to projects. If it is your first login the project table will be empty. Otherwise your projects show up here. To upload a new project in biom format just click the `Browse` Button and select the file.
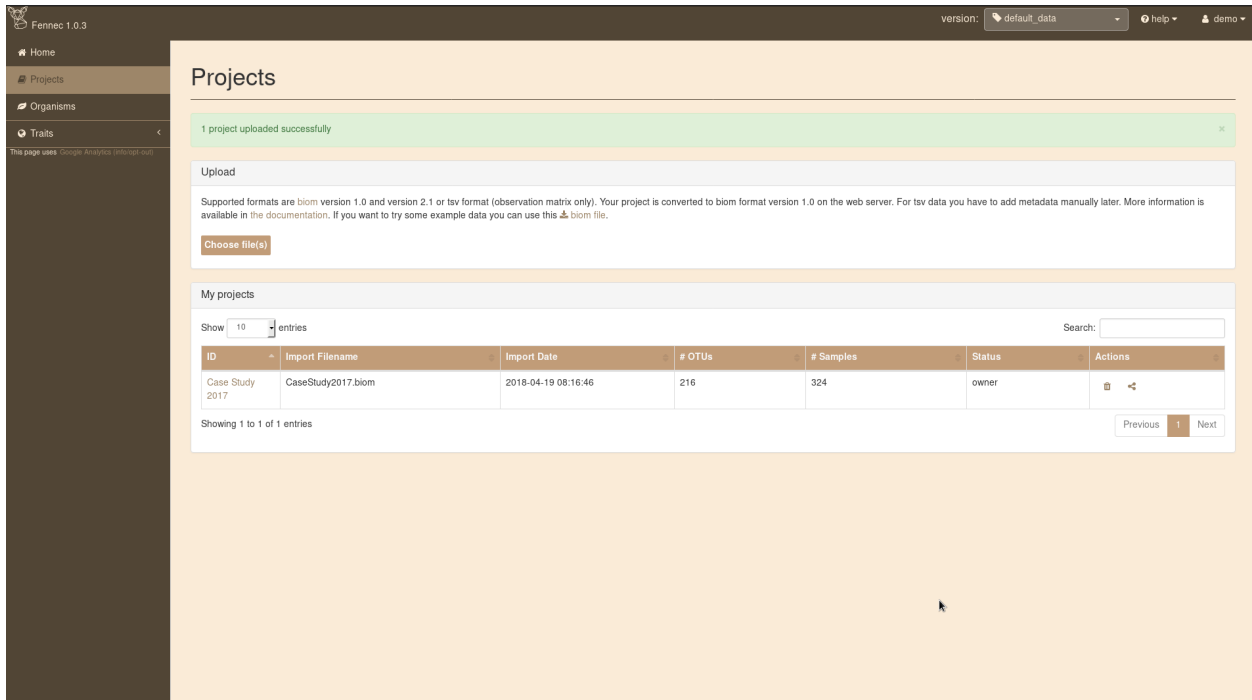
Details on file formats are available in the *Upload own data* section. Use this `demo biom file` to follow along the tutorial. It consists of 324 samples of pollen with a total of 216 OTUs.

To get to the project page click on the link in the first column of the project table *Case Study 2017*. This will bring you to the project details page.
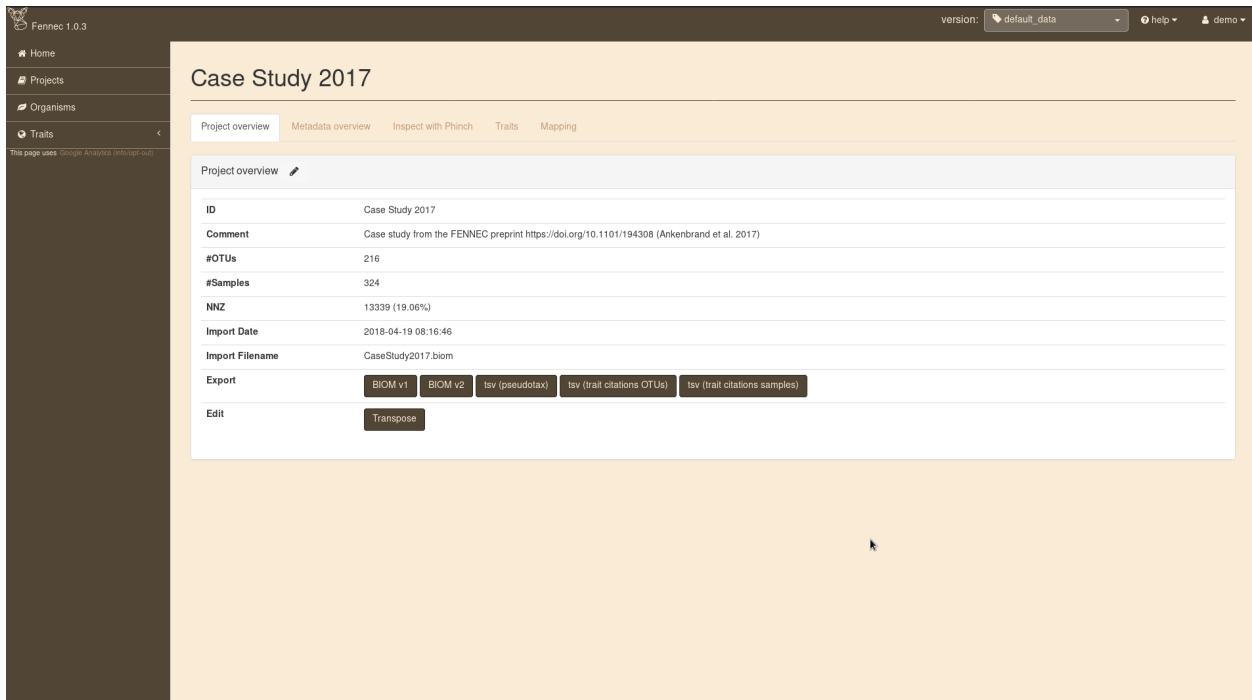
Basic information about the project is displayed in the table. However, when you navigate on the traits tab you'll see

empty tables. This is because the organisms in the project are not mapped to entities in the database, yet. So head to the mapping tab and select:

Map `OTUs` by `ncbi_taxid` on `NCBI taxid` and click `GO`.

The result of the mapping will be displayed below where a click on `Save in database` will make this mapping persistent.

Now that the OTUs are mapped to organisms in the database switching to the `Traits` tab will show available traits. By clicking on the icon in the details column for a trait (e.g. `Plant Habit`) you come to a page summarizing the trait values in this community. On this page trait values of all species present in any of the samples are summarized (without taking abundance into account). The trait values can be added to the project either via the `Add trait to OTU metadata` button on this page or the + icon in the trait table.

Note: When you add a trait to your project all trait entries from the Fennec database for this trait type and all organisms in the project are retrieved. If there are multiple values for a single organism they are collapsed (e.g. for categorical traits "red", "blue", "red" → "blue/red", for numerical traits the mean is used: 3.0, 5.0 → 4.0) Then this aggregated value is added as metadata with the trait type as key. Additionally, all individual citations are added to the "trait_citation" metadata key.
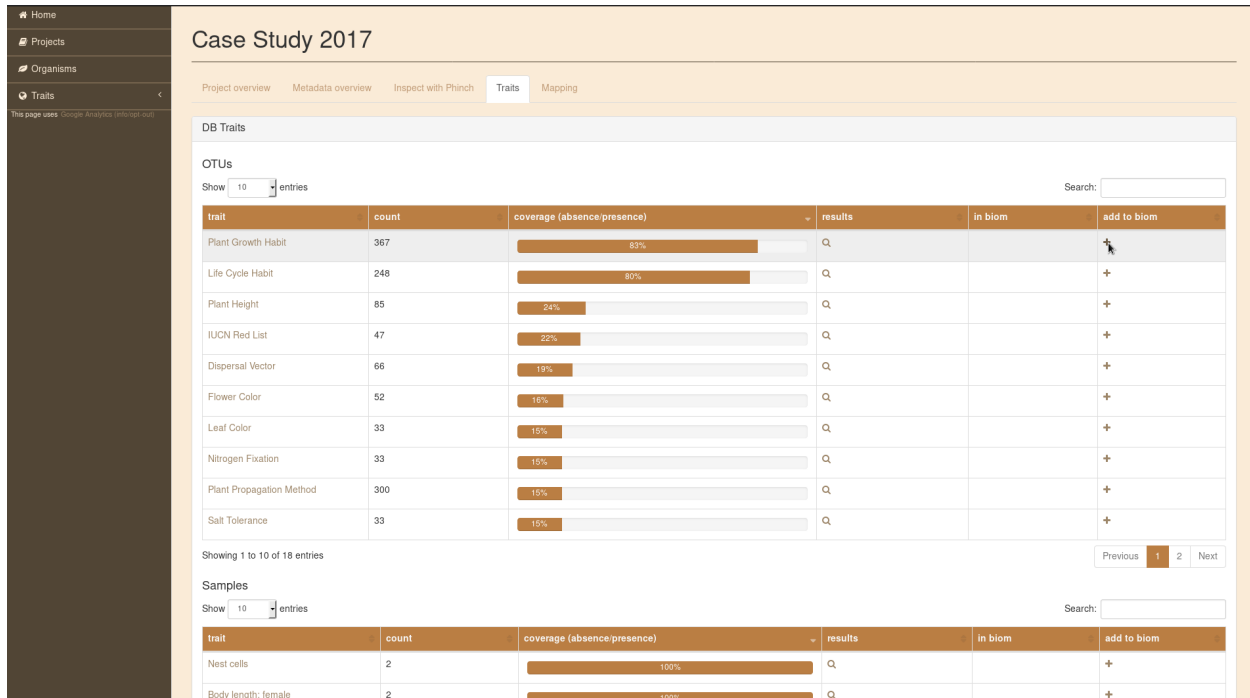
Finally it is possible to interactively explore the trait values by navigating to the `Inspect with Phinch` tab. The first screen on this tab is the filter page which you can leave via the `Proceed To Gallery` button in the top right corner. You can select any of the visualizations. Taxonomy is derived from your original biom file and not altered by Fennec. The traits you added can be selected in the top right corner of each visualization (except in the Sankey Diagram).
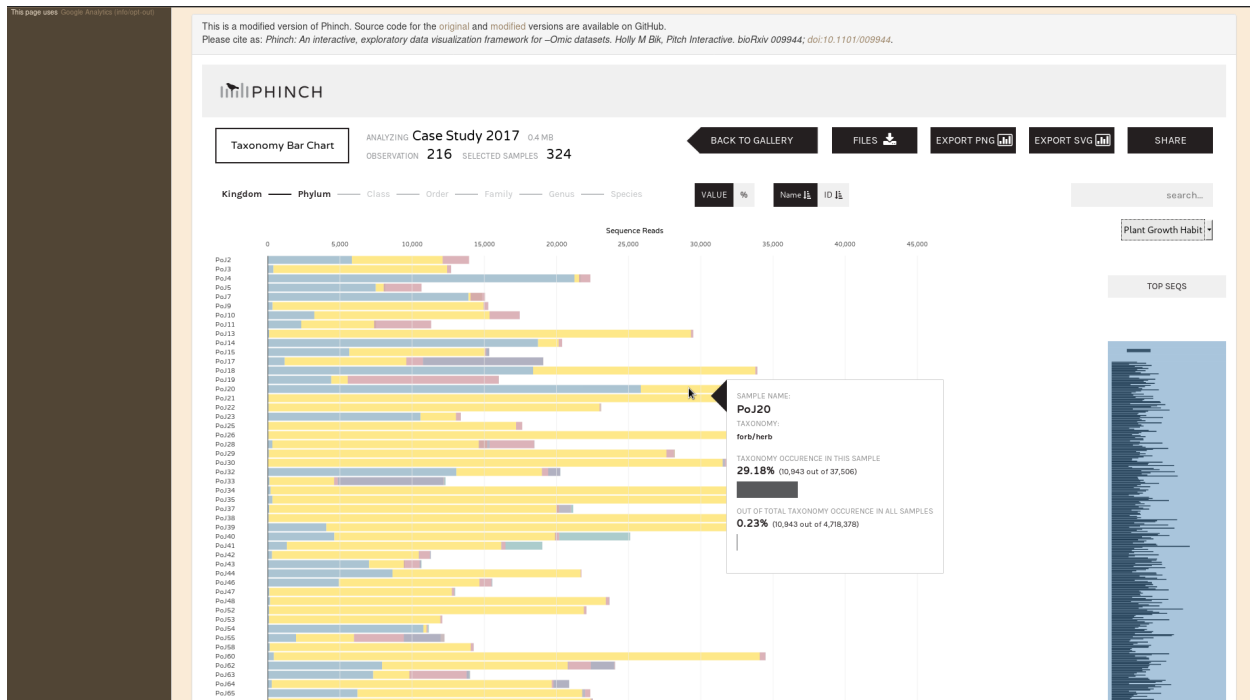
Download of the enriched community data is possible in different formats via the `Project overview` tab. For a more in depth analysis of this dataset see the *Pollination Ecology*.

2.2 Upload own data

It is possible to upload your own projects into FENNEC. By default those projects are only visible to you. There are a couple of features corrently in development that will enhance management of your own data, including upload of own traits and sharing of projects.

2.2.1 Projects

Internally FENNEC stores projects in BIOM (1.0) format. By integrating the biom-conversion-server it is also possible to upload projects in BIOM 2.x and even tabular (tsv) format.

BIOM

Upload in BIOM format should be straight forward. After login navigate to the 'Projects' page and on the top of the page hit the 'Browse' button under 'Upload projects'. There select one or more biom files (can be 1.0, 2.x, or a mixture). After you close the file selection dialog a message will appear, telling you '1 project uploaded successfully' and the project(s) will show up in the table.

You can upload BIOM files containing all of your metadata (e.g. files previously exported from FENNEC). Alternatively, you can also upload BIOM files that only contain your observation matrix and add metadata later. More details about the latter approach is given in the next section.

Table (tsv)

It is also possible to upload the observation matrix of a project in tsv (tab separated values) format. No metadata can be included in this table, instead all metadata needs to be added afterwards.

The process of uploading the observation tsv file and adding metadata is demonstrated with an example (Warning: no real data, do not look for biological signals). Given your observation matrix stored in a file called `otu_table.tsv` looks like this:

```
#OTUId      Sample1 Sample2 Sample3 Sample4
OTU1        33      76      12      8
OTU2        50      44      17      23
OTU3        9       99      15      55
OTU4        1       63      39      54
```

After normal upload on the project page it will appear in the table with ID 'None'. When you click the project you will see on the details page that upload was successful (you have 4 OTUs and 4 Samples). You can use the little pen icon beside 'Project overview' to edit the 'ID' and 'Comment' for the project.

However, as it is you can not do much with this project. To use the traits from FENNEC you need some way to map your OTUs (and/or samples) to organisms in the database. For this purpose we need to add metadata. So, assuming we have the following metadata files (also in tsv format), `otu_metadata.tsv`:

```
#OTUId      Scientific Name
OTU1        Bellis perennis
OTU2        Centaurea cyanus
OTU3        Medicago sativa
OTU4        Plantago lanceolata
```

and `sample_metadata.tsv`:

```
#Sample     Bee        Season
Sample1     Osmia caerulescens      Spring
Sample2     Megachile rotundata     Spring
Sample3     Osmia caerulescens      Summer
Sample4     Megachile rotundata     Summer
```

We can click the 'Metadata overview' tab and there we can add observation and sample metadata by sequentially clicking the 'Browse' buttons and selecting our respective tsv files. In both cases you should see a green message 'Successfully added metadata'. Now you can proceed by clicking 'Explore Sample metadata' and 'Explore OTU metadata' to see that both were added successfully.

Attention: By default the rows are ordered by 'Total Count' and not by 'ID'.

If we look on the 'Traits' tab we see, that both tables are still empty. But now we can head to the 'Mapping' tab and connect our OTUs and Samples with the corresponding FENNEC organisms in the database. First select:

Map `OTUs` by `Scientific Name` on `Scientific name`

and hit `GO`. If only `ID` is available in the second drop down, please reload the page. The Result should be that all 4 organisms have a scientific name and could be mapped to fennec_ids. So click the `Save in database` button to permanently store this mapping. After that the page should reload automatically and the bar for OTUs should show 100% mapped. Repeat those steps with `Samples` by `Bee` on `Scientific name` (do not forget to `Save in database`).

Now the tables on the 'Traits' tab are properly populated. You can inspect them and add them to your project. Lets add a couple of trais (using the + icon) for the OTUs and Samples

OTUs:

- Plant Growth Habit
- Life Cycle Habit
- Dispersal Vector

Samples:

- Sex ratio
- Foraging mode

Now it is time to 'Inspect with Phinch'. You can accept the default filters by clicking the 'Proceed to Gallery' button in the top right corner. Now select some visualizations and play around with the settings.

Attention: The Sankey Diagram as well as the taxonomic rank selection in Phinch will not work. This is because Phinch expects taxonomy information to be available as metadata in the 'taxonomy' field for OTUs. We did not provide this information with our metadata table and it is not automatically added by FENNEC. If you want to use this feature as well upload this `otu_taxonomy.tsv` file as additional OTU metadata. Don't worry, new metadata is added and will only overwrite existing metadata if it uses the same column name.

2.2.2 Traits

Coming soon.

Note: You can already load project specific metadata using the `Metadata overview` tab on the project details page.

2.3 Case Studies

This section shows possible use cases of Fennec by demonstrating the analysis of real world datasets. More case studies are always welcome so if you want to add one feel free to suggest one. We will gladly assist you in preparing and integrating everything (contact: markus.ankenbrand@uni-wuerzburg.de).

2.3.1 Pollination Ecology

This case study uses data from Sickel et al. 2015. This is the data set also used for the *Quick start* guide. Use this `biom file` to reproduce this analysis.

Introduction

In this study 384 pollen samples collected by two closely related solitary bee species of the Megachilidae were analyzed using next-generation sequencing, *Osmia bicornis* and *Osmia truncorum* (synonym *Heriades truncorum*). One of the bee species, *O. bicornis* is known to be polylectic, while the other, *O. truncorum* is oligolectic (focusing on Asteraceae). Although the data originates from next-generation sequencing, any community/network data can be used for the workflow independent of the method for data acquisition.

Three exemplary topics are to be addressed in this case study, with the first related to ecological interactions, followed by one concerning bio-monitoring and lastly one focusing on the socio-economic relevance:

1. **Are the two bee species showing preferences and differences between each other in growth habit types of visited plants?** Given the specialization of *O. truncorum* on Asteraceae (mainly forbs and herbs) one could hypothesize that this bee does not collect pollen from shrubs or trees. *O. bicornis* on the other hand collects from many different taxonomic plant groups. Is this reflected by a variety of growth habits or is there a specialization on plants of a specific growth habit, likewise to the other bee species? This hypotheses address the concept of a correlation between functional and taxonomic diversity of the visited plants.

2. **How many (and which) invasive species can be found in the samples? Are there vulnerable species in the samples? Is the amount of invasive and vulnerable species visited similar in all of the samples and by both species?** Monitoring the ranges of invasive as well as threatened plant species is an important task in conservation (Darling et al. 2007, Stout et al. 2009). Using pollen data collected by bees, presence of both types can be monitored by mapping conservation relevant traits to the network data. Further, pollination services by the bee species to both types can be identified.

3. **Which plants visited by the bees show agricultural relevance to humans and what is their relative amount compared to the remaining plant species?** Bees provide pollination services to agriculturally relevant plants (Bosch and Kemp 2002, Gruber et al. 2011, Klatt et al. 2013). Using traits such as *agricultural usage* allows to identify how specific the respective bees were in visiting such plants. On the other hand, solitary bees are important agents to ensure the pollination of wild plant species (Garibaldi et al. 2013), and using these traits it can be monitored whether the bees are mainly attracted to mass flowering crops or also visit other plants in agriculturally shaped landscapes.

Methods

The data has been downloaded from EBI-SRA project number PRJEB8640 and data preparation as well as taxonomic classification has been performed based on Sickel et al. 2015. The full workflow has been deposited at https://github.com/molbiodiv/meta-barcoding-dual-indexing . This resulted in a table with 1002 plant operational taxonomic units (OTU) and a total count of 6,979,584 observations (sequence reads). For each OTU, the taxonomic lineage and NCBI-taxonomy-ID have been determined during this process by hierarchic taxonomic assignments using UTAX (part of usearch, Edgar 2010). OTUs with total count of less or equal than 50 across all samples were excluded from the analysis. Samples with less than 10,000 sequence reads remaining have been removed as well. Finally, the remaining 353 plant OTUs were combined if they corresponded to the same taxon. The resulting table consists of 216 plant OTUs and 324 samples, which was imported into the Fennec. The total number of reads in this final dataset is 6,663,014. For the plants, the obtained NCBI-taxonomy-ID was used to map the OTUs in the community to organisms in the Fennec database, which resulted in all 216 OTUs being successfully mapped. For the samples, the corresponding bee species were mapped by the scientific name in the meta-data field "beeSpecies".

In the next step, values for "Plant Growth Habit", "EPPO Categorization", "World Crops Database", and "IUCN Red List" have been added to the project from the database. This dataset including the traits has then been interactively visualized and analyzed using the built-in modified version of Phinch (Bik et al. 2014) according to the research questions described above. Finally, the enriched dataset has been exported and imported into R using shiny-phyloseq (McMurdie et al. 2015) to demonstrate the usability of mapped data in further analyses tools. In particular a DCA ordination has been calculated and visualized with colorization by the trait "Plant Growth Habit". For this purpose OTUs with missing trait values and those with rare variants (keeping only forb/herb, tree, subshrub, shrub/tree, forb/herb/subshrub, forb/herb/vine) were filtered.

Results and Discussion

To show the potential of the Fennec to be used in ecological analysis, we conducted a case study as proof-of-concept for a pollen meta-barcoding data. We address multiple ecological questions and highlights some use cases, where automatic integration of public trait data with the FENNEC has been performed.

Are the two bee species showing preferences and differences between each other in growth habit types of visited plants?

A breakdown of the trait "Plant Growth Habit" for the two bee species separately (visualized via "Donut Partition Chart") reveals that for *O. truncorum* 89% of the taxonomic assignments were mappable to the trait, which resulted in the dominance of "forb/herb" with 87%. This matches our expectations as this bee is specialized on Asteraceae which mostly show this habit. For *O. bicornis*, 95% of the sequence data was assignable to "Plant Growth Habit", also with "forb/herb" with 65% being the most abundant, but a still considerable amount of 24% as "tree". Likewise to taxonomic specialization, no indication for a specialization on a specific plant growth habit is apparent. Another interesting observation is the trait coverage when taking abundance into account. While only 85% of OTUs have a value for "Plant Growth Habit", those OTUs contribute 93% to the entire community. Thus the OTUs with missing traits are relatively rare in the community, with the more abundant ones being well-studied. Automatically-mapped trait data also helps in interpretation of beta-diversity turnover between samples (pollens). For example, ordinations can be visualized with trait data, in our case "Plant Growth Habit", as a split-plot with samples shaped by bee species and plant taxa colored by Plant Growth Habit. In our case study, samples are separated by bee species as expected on the first ordination axis with all samples from *O. truncorum* mostly isolated on the right-hand side. OTUs localized similarly with possible values for ordination axis 1 were almost exclusively forbs and herbs. The variation of traits for plants visited by this bee species on the second axis is negligible. For *O. bicornis* there is a substantial spread on the second axis, where plants of type *tree* seem to concentrate in the upper part. The trait data helps to understand the ecology behind the dataset, indicating plant turnover and eventually also location and landscape changes to be represented on the second axis.

How many (and which) invasive species can be found in the samples? Are there vulnerable species in the samples? Is the amount of invasive and vulnerable species visited similar in all of the samples and by both species?

Fig. 1: Splitplot of a DCA ordination. Samples are in the left panel with shapes according to by bee species. OTUs are in the right panel with points colored by growth habit (filtered for most common growth habits, species with missing trait have been removed). Samples split nicely by bee on the first axis with *O. truncorum* on the right-hand side. The OTUs on the right-hand side of the ordination are as expected mainly forb/herb. For *O. bicornis* there is a substantial spread on the second axis.

The trait "EPPO Categorization" was mapped to our pollen collection data to determine if and to what extent the samples contain species that are regarded as invasive in Europe. One of the visualization methods of the Phinch suite that is integrated into the Fennec the "Bubble Chart", has been applied to explore this trait. It reveals three samples containing high numbers of invasive species (PoJ74, PoJ236, PoJ244). Further inspection with the integrated meta-data tables showed that PoJ74 and PoJ244 have more than 1000 counts of *Solidago canadensis*, each while PoJ236 has a count of 2779 for *Helianthus tuberosus*. So Fennec can be used to find samples with high amounts of invasive species and their corresponding geographical locations (if they are part of the sample metadata). It might thus serve as indicator for occurrence of invasive species in geographic regions and used to monitor the spread of invasive species over space and time. Regarding the occurrence of species with respect to threat status, the pollen data was automatically mapped to the IUCN red list data and the distribution of vulnerable species (as listed by the IUCN) across samples was visualized using the "Bubble Chart", but also a "Taxonomy Bar Chart". These charts illustrate that multiple samples consist almost entirely of "near threatened" species, particularly *Juglans regia*, the english walnut, which experienced strong declines through anthropogenic overuse and lack of replacement plantings. As indicated by the data, it served as a major nutrient source for individual investigated bees.

Which plants visited by the bees are agriculturally relevant to humans and what is their relative amount compared to the remaining plant species?

Finally ecologists (especially in the field of conservation) are often in the difficult situation of having to quantify economic value of ecosystem services like pollination (Hanley et al. 2015). The Fennec helps in addressing such socio-economic questions by including human usage (as crop) as a trait. All plants listed in the World Crops Database are known to be cultivated by humans for specific purposes. The "Donut Partition Chart" for this trait reveals that 36.7% of plants collected by *O. bicornis* and 7.3% of plants collected by *O. truncorum* are listed in that database. This does not yet give more information like the category of crop (e.g. fruits, vegetables, nuts, wood product, etc.) or a real monetary quantification. However this is not a limitation of Fennec but of the underlying data (i.e. if this data is available it can be imported into Fennec and is then automatically available for the community of interest).

Fig. 2: Partition donut charts for the trait "World Crops Database" separated by bee species. Plot has been created with the built in modified version of Phinch.

2.3.2 Pollination Network

Coming soon. . .

This case study uses data from Bell et al. 2017. Use this `biom file` to reproduce this analysis.

2.3.3 Microbiome Study

Coming soon. . .

This case study uses data from Song et al. 2013. Use this `biom file` to reproduce this analysis.

Admin manual

This section of the manual describes the process of setting up your own instance of Fennec. It explains how to configure it and how to load data to the database. If you are looking for a manual on how to use an existing Fennec instance please refer to *User manual*. If you want to extend or enhance Fennec have a look at the README in the repository.

3.1 Docker setup

Note: In order to make setup of new instances as easy as possible we describe the setup using docker compose. If you do not want to use docker compose it is possible to do it with plain docker or even without docker. Feel free to open an issue if you encounter any problems.

Install docker and docker compose according to the documentation. Now create a folder for the fennec instance on the target machine and download the docker compose file:

```
mkdir fennec
cd fennec
wget https://raw.githubusercontent.com/molbiodiv/fennec/master/docker/fennec/docker-
↪compose.yml
# Get initial versions of the main configuration file and the contact page
wget -O parameters.yml https://raw.githubusercontent.com/molbiodiv/fennec/master/app/
↪config/parameters.yml.dist
wget -O custom_contact.html.twig https://raw.githubusercontent.com/molbiodiv/fennec/
↪master/app/Resources/views/misc/missing_contact.html.twig
# Create empty data folder with correct owner
mkdir data
```

Note: The name of the folder is relevant because docker compose will use this as the project name. If you want to have multiple fennec instances on the same host make sure to use different directory names. In the following it is

assumed that `docker-compose` is always executed from inside your fennec directory.

Have a look at the `docker-compose.yml` file and edit it as needed (e.g. adjust the port you want to use). Another important thing to note is that by default the web image is `iimog/fennec` which is automatically build from the `master` branch of the fennec repository on GitHub. Therefore, this image might contain changes that are not yet part of an official stable release. If you want to have a specific Fennec version instead you can add that version to the image, like `iimog/fennec:v1.0.3` or to get the latest development version you can use the tag develop, like `iimog/fennec:develop`. Now it is time to create and initialize the fennec instance:

```
docker-compose up -d
# wait a couple of seconds to allow the databases to boot
# Now initialize the userdb and the default_data db
docker-compose exec web /fennec/bin/console doctrine:schema:create --em userdb
docker-compose exec web /fennec/bin/console doctrine:schema:create --em default_data
docker-compose exec web /fennec/bin/console doctrine:fixtures:load --em default_data -
→n
```

Congratulations, Fennec is now running on http://localhost . However, Fennec does not contain any data, yet.

3.2 Configuration

3.2.1 Create admin user

The admin user is able to manage other users and needs to be created via the command line. You can choose the username, email and password freely. If you do not provide the password as last argument you will be prompted for it. This avoids adding this sensible information to your command history. There will be no visual feedback while you type the password:

```
docker-compose exec web /fennec/bin/console app:create-user --super-admin <username>
→<email> [password]
```

Note: If you forget the password of the admin user you can create a new one and use the admin web interface to edit or delete the old account.

3.2.2 Login with GitHub

1. Register an OAuth App with your account following [this guide](https://developer.github.com/apps/building-integrations/setting-up-and-registering-oauth-apps/)

2. As "Authorization callback URL" enter your domain or ip address with `/login` appended

3. Modify `parameters.yml` and add the respective values to `github_client_id` and `github_client_secret`

That's it. Login with GitHub should now work.

Warning: After changes to `parameters.yml` it might be necessary to clear the cache:
```
docker-compose restart
docker-compose exec -u www-data web /fennec/bin/console cache:clear --env prod
docker-compose exec -u www-data web /fennec/bin/console cache:warmup --env prod
```

3.2.3 Contact Page

The content of `custom_contact.html.twig` is integrated into the contact page. You can modify it like this for example:

```
<div class="row">
    <h1>Contact</h1>
    This instance is maintained by <a href="mailto:mail@example.com">John Doe</a>.
    The source code is available on <a href="https://github.com/molbiodiv/fennec">
→GitHub</a>.
</div>
```

Please be aware that a proper contact page might be a legal requirement if you run a public instance.

3.2.4 Google Analytics

It is possible to monitor the traffic of your page with Google Analytics which is disabled by default. If you want to enable it make sure that you are allowed to do this legally, and then add your tracking id to `parameters.yml` as value for `ga_tracking`.

3.3 Loading organisms

The following sections describe in detail how to import organisms and traits into a Fennec database. Those are the commands used to import the `default_data` into the public instance. If you want to start with a mirror of this database without importing everything manually you can use `this dump`. Skip ahead to *Import database from dump*.

3.3.1 NCBI Taxonomy

We will demonstrate loading organisms into the `default_data` database using NCBI Taxonomy. Inside the docker container execute the following commands:

```
curl ftp://ftp.ncbi.nih.gov/pub/taxonomy/taxdump.tar.gz >data/taxdump.tar.gz
tar xzvf data/taxdump.tar.gz -C data
grep "scientific name" data/names.dmp | perl -F"\t" -ane 'print "$F[2]\t$F[0]\n"' >
→data/ncbi_organisms.tsv
docker-compose exec web /fennec/bin/console app:import-organism-db --provider ncbi_
→taxonomy /data/ncbi_organisms.tsv
```

The last step will take a couple of minutes but after that more than 1.7 million organisms will be stored in the database with their scientific name and NCBI taxid.

> **Attention:** The taxonomy is currently only used to display it on the organism page. There are possible future applications like automatic trait imputation based on taxonomy. However, none of them are implemented, yet. Therefore, you might consider not importing taxonomic information, especially as the import is quite cumbersome. If taxonomic information is used more in FENNEC the import process will be improved as well. For now the steps below are required.

In order to add taxonomic relationships follow those steps:

```
# Create a fennec_id to ncbi_taxid map (will be obsolete in the future)
docker-compose exec -T datadb psql -U fennec_data -F $'\t' -At -c "SELECT fennec_id,
↪identifier as ncbi_taxid FROM fennec_dbxref, db WHERE fennec_dbxref.db_id=db.id AND
↪db.name='ncbi_taxonomy';" >data/fennec2ncbi.tsv
perl -F"\t" -ane 'BEGIN{open IN, "<data/fennec2ncbi.tsv";while(<IN>){chomp;($f,
↪$n)=split(/\t/);$n2f{$n}=$f}} print "$n2f{$F[0]}\t$n2f{$F[2]}\t$F[4]\n"' data/nodes.
↪dmp >data/ncbi_taxonomy.tsv
wget -P data https://raw.githubusercontent.com/molbiodiv/fennec-cli/master/bin/import_
↪taxonomy.pl
docker-compose exec web perl /data/import_taxonomy.pl --input /data/ncbi_taxonomy.tsv
↪--provider ncbi_taxonomy --db-host datadb --db-user fennec_data --db-password
↪fennec_data --db-name fennec_data
```

Again the last step will take some minutes (even after printing "Script finished") and needs a few GB of memory.

3.3.2 EOL

The Encyclopedia of Life is a great resource for organism information. Because of the nice API organism pages in Fennec are dynamically created from EOL content. In order to link organisms to EOL we need to add EOL page IDs. For this purpose we use the hierarchy entries file:

```
wget -P data http://opendata.eol.org/dataset/da9635ec-71b6-4fb2-a4cb-518f71eeb45d/
↪resource/dd1d5160-b56a-4541-ac88-494bc03b4bc8/download/hierarchyentries.tgz
tar xzvf data/hierarchyentries.tgz -C data
# Now we create a file with two columns: 1) ncbi_taxid 2) eol_id
perl -F"\t" -ane 'print "$F[4]\t$F[1]\n" if($F[2] == 1172)' data/hierarchy_entries.
↪tsv | perl -pe 's/"//g' | sort -u >data/ncbi2eol.tsv
docker-compose exec web php -d memory_limit=2G /fennec/bin/console app:import-
↪organism-ids --provider EOL --mapping ncbi_taxonomy --skip-unmapped /data/ncbi2eol.
↪tsv
```

Now you have 1.7 million organisms in the database of which roughly 1.2 million have a nice organism page provided by EOL.

3.4 Loading traits

3.4.1 Plant Growth Habit

As a first example we want to load growth habit data for plants from eol. Those values are stored in this file from opendata.eol.org:

```
wget -P data https://editors.eol.org/eol_php_code/applications/content_server/
↪resources/eol_traits/growth-habit.txt.gz
gunzip data/growth-habit.txt.gz
# We want to have a tsv with the following columns: eol_id, value, value_ontology,
↪citation, origin_url
# And citation consists of the columns "Supplier(12),Citation(15),Reference(29),
↪Source(14)"
perl -F"\t" -ane 'print "$F[0]\t$F[4]\t$F[6]\tSupplier:$F[12];Citation:$F[15];
↪Reference:$F[29];Source:$F[14]\t$F[13]\n" unless(/^EOL page ID/)' data/growth-habit.
↪txt >data/growth-habit.tsv
```

(continues on next page)

```
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "general growth form, including size and branching. Some_
→organisms have different growth habits depending on environment or location" --
→ontology_url "http://www.eol.org/data_glossary#http___eol_org_schema_terms_
→PlantHabit" "Plant Growth Habit"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Plant Growth Habit" --provider TraitBank --description "EOL_
→TraitBank http://eol.org/info/516" --mapping EOL --skip-unmapped --public --default-
→citation "Data supplied by Encyclopedia of Life via http://opendata.eol.org/ under_
→CC-BY" /data/growth-habit.tsv
```

Almost 70 thousand of the entries are imported into the database. For the other EOL ids there is no organism in the database, therefore those are skipped (because of the `--skip-unmapped` parameter, otherwise the importer would fail).

An important thing to note is that we are preparing the trait table by rearranging columns using `perl`. However, you could just as well use `Excel` or any other tool to do this. The only requirement is that you end up with a tab delimited file with five columns:

1. organism identifier (either fennec_id or something that can be mapped)

2. trait value

3. value ontology url (can be empty)

4. citation (can be empty or set via default citation, if multiple sources have to be cited they have to be concatenated)

5. origin url (can be empty, a link to the original source)

3.4.2 Life Cycle Habit

Next we can repeat these steps for the "Life Cycle Habit" trait: Again there is a file at opendata.eol.org:

```
wget -P data http://opendata.eol.org/dataset/fedb8890-f943-4907-a36f-c7df4770a076/
→resource/e4eced0b-70f4-497f-9aa6-b1fd1212cfd9/download/life-cycle-habit.txt.gz
gunzip data/life-cycle-habit.txt.gz
perl -F"\t" -ane 'print "$F[0]\t$F[4]\t$F[6]\tSupplier:$F[12];Citation:$F[15];
→Reference:$F[29];Source:$F[14]\t$F[13]\n" unless(/^EOL page ID/)' data/life-cycle-
→habit.txt >data/life-cycle-habit.tsv
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Determined for type of life cycle being annual, binneal,_
→perennial etc." --ontology_url "http://purl.obolibrary.org/obo/TO_0002725" "Life_
→Cycle Habit"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Life Cycle Habit" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/life-cycle-habit.tsv
```

3.4.3 EPPO List of Invasive Alien Plants (Europe)

The European and Mediterranean Plant Protection Organization (EPPO) provides a list of invasive alien species: https://www.eppo.int/INVASIVE_PLANTS/ias_lists.htm This categorizations can be obtained as csv file from: https://gd.eppo.int/rppo/EPPO/categorization.csv In order to import this file into FENNEC execute those commands in the docker container:

```
curl "https://gd.eppo.int/rppo/EPPO/categorization.csv" >data/eppo_categorization.csv
perl -pe 's/"//g' data/eppo_categorization.csv | perl -F"," -ane 'print "$F[3]\t
→$F[1]\t\tEPPO (2017) EPPO Global Database (available online). https://gd.eppo.
→int\thttps://gd.eppo.int/rppo/EPPO/categorization.csv\n" if($F[6]=="")' >data/eppo_
→categorization.tsv
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "List of invasive alien species by the European and␣
→Mediterranean Plant Protection Organization (EPPO)" --ontology_url "https://www.
→eppo.int/INVASIVE_PLANTS/ias_lists.htm" "EPPO Categorization"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "EPPO Categorization" --provider EPPO --description "European␣
→and Mediterranean Plant Protection Organization (EPPO) https://www.eppo.int/" --
→mapping scientific_name --skip-unmapped --public --default-citation "EPPO (2017)␣
→EPPO Global Database (available online). https://gd.eppo.int" /data/eppo_
→categorization.tsv
```

3.4.4 World Crops Database

The World Crops Database is a collection of cereals, fruits, vegetables and other crops that are grown by farmers all over the world collected by Hein Bijlmakers at http://world-crops.com/ . It has a list of plants by scientific name http://world-crops.com/showcase/scientific-names/ which can be used for import into FENNEC. Being on this list is a strong indication that the plant can be used for agriculture. The definition of crop used for the database is: "Agricultural crops are plants that are grown or deliberately managed by man for certain purposes." (see http://world-crops.com/the-world-crops-database/) To prepare the data for import into FENNEC (just the info that a plant is listed) execute:

```
# Citation will be provided as default citation (therefore left empty here)
curl "http://world-crops.com/showcase/scientific-names/" | grep Abelmoschus | perl -
→pe 's/\|/\n/g;s/.*a href="([^"]+)" >([^<]+).*/$2\tlisted\t\t\t$1/g' | grep -v "</p>
→" | sort -u >data/crops.tsv
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "The World Crops Database is a collection of cereals, fruits,␣
→vegetables and other crops that are grown by farmers all over the world. In this␣
→context crops are defined as 'Agricultural crops are plants that are grown or␣
→deliberately managed by man for certain purposes.'" --ontology_url "http://world-
→crops.com/" "World Crops Database"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --provider WorldsCropDatabase --description "The World Crops Database http:/
→/world-crops.com/the-world-crops-database/" --default-citation "Hein Bijlmakers,␣
→'World Crops Database', available online http://world-crops.com/showcase/scientific-
→names/ (retrieved $(date "+%Y-%m-%d"))" --traittype "World Crops Database" --
→mapping scientific_name --skip-unmapped /data/crops.tsv
```

The database also contains categories like Vegetables, Cereals, Fruits, etc. So in principle those categories could be used as value instead of a generic "listed".

3.4.5 More TraitBank plant traits

A couple more interesting plant traits from TraitBank are available at http://opendata.eol.org/dataset/plantae This dataset consists of thirteen traits:

- conservation status (will not be imported because we use IUCN directly)

- dispersal vector

- flower color

- invasive in
- leaf area
- leaf color
- nitrogen fixation
- plant height
- plant propagation method
- salt tolerance
- soil pH
- soil requirements
- vegetative spread rate

Three of them are numerical (leaf area, plant height, and soil pH) they are discussed in the next section. In order to create the categorical trait types and import them into FENNEC just follow the steps below:

```
# Download and prepare data
wget http://opendata.eol.org/dataset/a44a37ad-27f5-45ef-8719-1a31ae4ed3e5/resource/
↪c7c90510-402e-4ead-8204-d92c44723c1f/download/plantae.zip -O data/plantae.zip
unzip data/plantae.zip -d data
wget http://opendata.eol.org/dataset/a44a37ad-27f5-45ef-8719-1a31ae4ed3e5/resource/
↪67410c56-d9d9-4e60-a223-39334e0081d5/download/uses.txt.gz -O data/Plantae/Plantae-
↪uses.txt.gz
for i in data/Plantae/*.txt.gz
do
    BASE=$(basename $i .txt.gz)
    zcat $i | perl -F"\t" -ane 'print "$F[0]\t$F[4]\t$F[6]\tSupplier:$F[12];Citation:
↪$F[15];Reference:$F[29];Source:$F[14]\t$F[13]\n" unless(/^EOL page ID/)' >data/
↪$BASE.tsv
done


# Create trait types (description and ontology url from http://eol.org/data_glossary )
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
↪free --description "A dispersal vector is an agent transporting seeds or other_
↪dispersal units. Dispersal vectors may include biotic factors, such as animals, or_
↪abiotic factors, such as the wind or the ocean." --ontology_url "http://eol.org/
↪schema/terms/DispersalVector" "Dispersal Vector"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
↪free --description "A flower anatomy and morphology trait (TO:0000499) which is_
↪associated with the color of the flower (PO:0009046)." --ontology_url "http://purl.
↪obolibrary.org/obo/TO_0000537" "Flower Color"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
↪free --description "Information about the jurisdictions where the taxon is_
↪considered to be an invasive organism due to its negative impact on human welfare_
↪or ecosystems." --ontology_url "http://eol.org/schema/terms/InvasiveRange"
↪"Invasive In"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
↪free --description "A vascular leaf anatomy and morphology trait (TO:0000748) which_
↪is associated with the color of leaf (PO:0025034)." --ontology_url "http://purl.
↪obolibrary.org/obo/TO_0000326" "Leaf Color"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
↪free --description "The process in which nitrogen is taken from its relatively_
↪inert molecular form (N2) in the atmosphere and converted into nitrogen compounds_
↪useful for other chemical processes, such as ammonia, nitrate and nitrogen dioxide.
↪" --ontology_url "http://purl.obolibrary.org/obo/GO_0009399" "Nitrogen Fixation"
```

```
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Methods used to produce new plants from a parent plant." --
→ontology_url "http://eol.org/schema/terms/PropagationMethod" "Plant Propagation_
→Method"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Tolerance to the high salt content in the growth medium." --
→ontology_url "http://purl.obolibrary.org/obo/TO_0006001" "Salt Tolerance"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "The soil requirements (texture, moisture, chemistry) needed for_
→a plant to establish and grow." --ontology_url "http://eol.org/schema/terms/
→SoilRequirements" "Soil Requirements"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "The rate at which this plant can spread compared to other_
→species with the same growth habit." --ontology_url "http://eol.org/schema/terms/
→VegetativeSpreadRate" "Vegetative Spread Rate"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "The uses of the organism or products derived from the organism.
→" --ontology_url "http://eol.org/schema/terms/Uses" "Uses"


# Import traits
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Dispersal Vector" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-dispersal-vector.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Flower Color" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-flower-color.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Invasive In" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-invasive-in.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Leaf Color" --provider TraitBank --mapping EOL --skip-unmapped_
→--public --default-citation "Data supplied by Encyclopedia of Life via http://
→opendata.eol.org/ under CC-BY" /data/Plantae-leaf-color.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Nitrogen Fixation" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-nitrogen-fixation.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Plant Propagation Method" --provider TraitBank --mapping EOL --
→skip-unmapped --public --default-citation "Data supplied by Encyclopedia of Life_
→via http://opendata.eol.org/ under CC-BY" /data/Plantae-plant-propagation-method.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Salt Tolerance" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-salt-tolerance.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Soil Requirements" --provider TraitBank --mapping EOL --skip-
→unmapped --public --default-citation "Data supplied by Encyclopedia of Life via_
→http://opendata.eol.org/ under CC-BY" /data/Plantae-soil-requirements.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Vegetative Spread Rate" --provider TraitBank --mapping EOL --
→skip-unmapped --public --default-citation "Data supplied by Encyclopedia of Life_
→via http://opendata.eol.org/ under CC-BY" /data/Plantae-vegetative-spread-rate.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Uses" --provider TraitBank --mapping EOL --skip-unmapped --
→public --default-citation "Data supplied by Encyclopedia of Life via http://
→opendata.eol.org/ under CC-BY" /data/Plantae-uses.tsv
```

By now you should have an idea on how importing categorical traits into FENNEC works.

3.4.6 Numerical Traits

> **Attention:** The numerical traits need a little more attention as there are two potential complications:
>
> 1. The values might have different units
>
> 2. The values might represent different kinds of statistics (single measurement, mean, median, min, max)
>
> Regarding 1: FENNEC associates a single unit for each trait type. Therefore all numbers have to be converted to this unit. Regarding 2: In order to allow simple usage of numerical values in community analyses FENNEC has no notion of those different types. Instead FENNEC treats all values for one organism identically and uses their mean to aggregate them. Therefore it is important to only import meaningful values (mean, median, in some cases measurements, in case of a symmetric distribution min and max together might make sense as well). This short coming could be fixed in the future by adding more fine grained trait formats (e.g. numerical-range)

To import the traits downloaded above in the plantae dataset from http://opendata.eol.org/dataset/plantae do this inside the docker container:

```
# data preparation
# For leaf area some values are numeric (unit mm^2 or cm^2) some categorical (large,
→medium, samll, ...) all methods are either measurement or average. Therefore all
→numeric values are used and converted to cm^2. Unit neads to be stripped from
→values.
zcat data/Plantae/Plantae-leaf-area.txt.gz | perl -F"\t" -ane 'BEGIN{%factor=("cm^2"
→=> 1, "mm^2" => 0.01)} $F[4]=~s/,//g;$F[4]=~s/ .*//g; print "$F[0]\t".($F[4] *
→$factor{$F[7]})."\t$F[6]\tSupplier:$F[12];Citation:$F[15];Reference:$F[29];Source:
→$F[14]\t$F[13]\n" unless(/^EOL page ID/ or $F[7] eq "")' >data/Plantae-leaf-area.tsv
# For plant height we convert all units (cm, ft, inch, m) to cm and discard rows that
→use statistical method http://semanticscience.org/resource/SIO_001114 (max),
→retaining average, median and measurement
zcat data/Plantae/Plantae-plant-height.txt.gz | perl -F"\t" -ane 'BEGIN{%factor=("cm"
→=> 1, "m" => 100, "ft" => 30.48, "inch" => 2.54)} print "$F[0]\t".($F[4] * $factor{
→$F[7]})."\t$F[6]\tSupplier:$F[12];Citation:$F[15];Reference:$F[29];Source:$F[14]\t
→$F[13]\n" unless(/^EOL page ID/ or $F[17] eq "http://semanticscience.org/resource/
→SIO_001114")' >data/Plantae-plant-height.tsv
# pH has no unit so that is not a problem. However the method here is either min or
→max. But we have both values for every EOL ID except 1114581 and 584907 (verify
→with zcat Plantae/Plantae-soil-pH.txt.gz | cut -f1,18 | sort -u | cut -f1 | sort |
→uniq -u ).
zcat data/Plantae/Plantae-soil-pH.txt.gz | perl -F"\t" -ane 'print "$F[0]\t$F[4]\t
→$F[6]\tSupplier:$F[12];Citation:$F[15];Reference:$F[29];Source:$F[14]\t$F[13]\n"
→unless(/^EOL page ID/ or $F[0] eq "1114581" or $F[0] eq "584907")' >data/Plantae-
→soil-pH.tsv


# Create trait types (incl. unit)
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "A leaf anatomy and morphology trait (TO:0000748) which is associated
→with the total area of a leaf (PO:0025034)." --ontology_url "http://purl.obolibrary.
→org/obo/TO_0000540" --unit "cm^2" "Leaf Area"
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "A stature and vigor trait (TO:0000133) which is associated with the
→height of a whole plant (PO:0000003)." --ontology_url "http://purl.obolibrary.org/
→obo/TO_0000207" --unit "cm" "Plant Height"
```

```
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
↪description "The soil pH, of the top 12 inches of soil, within the plant's known␣
↪geographical range. For cultivars, the geographical range is defined as the area to␣
↪which the cultivar is well adapted rather than marginally adapted." --ontology_url
↪"http://eol.org/schema/terms/SoilPH" "Soil pH"


# import
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
↪entries --traittype "Leaf Area" --provider TraitBank --mapping EOL --skip-unmapped -
↪-public --default-citation "Data supplied by Encyclopedia of Life via http://
↪opendata.eol.org/ under CC-BY" /data/Plantae-leaf-area.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
↪entries --traittype "Plant Height" --provider TraitBank --mapping EOL --skip-
↪unmapped --public --default-citation "Data supplied by Encyclopedia of Life via␣
↪http://opendata.eol.org/ under CC-BY" /data/Plantae-plant-height.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
↪entries --traittype "Soil pH" --provider TraitBank --mapping EOL --skip-unmapped --
↪public --default-citation "Data supplied by Encyclopedia of Life via http://
↪opendata.eol.org/ under CC-BY" /data/Plantae-soil-pH.tsv
```

This will import the numerical trait values into FENNEC. The count for "Distinct new values" will be displayed as 0 as this is specific for categorical values.

3.4.7 SCALES Wasps & Bees Database

This database (available at http://scales.ckff.si/scaletool/?menu=6&submenu=3) is an excellent resource for many traits of 162 bees and wasps. As data download is not easily possible here is a guide on downloading all the data and extracting the traits: First download the html pages of all organisms to an empty folder (sid ranges from 1 to 162, determined by trial and error):

```
mkdir -p data/scales
for i in $(seq 1 162)
do
    curl "http://scales.ckff.si/scaletool/index.php?menu=6&submenu=3&sid=$i" >data/
↪scales/$i.html
done
```

To extract all traits I wrote a short python script (using Beautiful Soup) available as gist. You can extract traits with those commands:

```
# Install beautiful soup (e.g. via "conda install beautifulsoup4")
cd data/scales
wget https://gist.githubusercontent.com/iimog/a6a36a7b03906f18ac490b0a4708224c/raw/
↪b3bc7309ae13415c9d00ad469e948b8847312511/extract_scales_bee_traits_from_html.py
python extract_scales_bee_traits_from_html.py
# Get rid of colon in filenames
rename 's/://g' *.tsv
# Osmia rufa and Osmia bicornis are synonyms but bicornis is used by NCBI taxonomy␣
↪while rufa is used by SCALES, therefore: rename globally:
perl -i -pe 's/Osmia rufa/Osmia bicornis/g' *.tsv
cd -
```

This will create a bunch of tsv files with categorical and numerical values for each trait as well as a file `trait_types.tsv` which lists all trait types with description. Using mapping by scientific name those files can be imported directly:

```
# Create trait types (incl. unit)
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "Average number of brood cells per nest" "Nest cells"
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "Approximate body length of female collection specimens" --unit "mm"
→"Body length: female"
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "Mean weight of a freshly hatched adult female" --unit "mg" "Adult␣
→weight: female"
docker-compose exec web /fennec/bin/console app:create-traittype --format numerical --
→description "Male/female rate of progeny" "Sex ratio"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Sex ratio categories: female biased (males/females<0.8), equal␣
→(males/females 0.8-1.3), male biased (males/females>1.3)" "Sex ratio (categorical)"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free "Larval food type"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free "Foraging mode"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Typical of a landscape species" "Landscape type"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Nest building material type" "Nest built of"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Trophic specialisation rank" "Trophic specialisation"
docker-compose exec web /fennec/bin/console app:create-traittype --format categorical_
→free --description "Taxonomic rank on which this organism is specialized on"
→"Specialized on"


# import
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Nest cells" --provider SCALES_WaspsBeesDatabase --description
→"SCALES Wasps & Bees Database http://scales.ckff.si/scaletool/?menu=6&submenu=3" --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Nest cells_numeric.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Body length: female" --provider SCALES_WaspsBeesDatabase --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Body length female_numeric.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Adult weight: female" --provider SCALES_WaspsBeesDatabase --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Adult weight female_numeric.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Sex ratio" --provider SCALES_WaspsBeesDatabase --mapping␣
→scientific_name --skip-unmapped --public --default-citation "Budrys, E., Budriene.,␣
→A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/data/scales/
→Sex ratio_numeric.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Sex ratio (categorical)" --provider SCALES_WaspsBeesDatabase --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Sex ratio_categorical.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Larval food type" --provider SCALES_WaspsBeesDatabase --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Larval food type_categorical.tsv"
```

```
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Foraging mode" --provider SCALES_WaspsBeesDatabase --mapping␣
→scientific_name --skip-unmapped --public --default-citation "Budrys, E., Budriene.,␣
→A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/data/scales/
→Foraging mode_categorical.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Landscape type" --provider SCALES_WaspsBeesDatabase --mapping␣
→scientific_name --skip-unmapped --public --default-citation "Budrys, E., Budriene.,␣
→A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/data/scales/
→Landscape type_categorical.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Nest built of" --provider SCALES_WaspsBeesDatabase --mapping␣
→scientific_name --skip-unmapped --public --default-citation "Budrys, E., Budriene.,␣
→A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/data/scales/
→Nest built of_categorical.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Trophic specialisation" --provider SCALES_WaspsBeesDatabase --
→mapping scientific_name --skip-unmapped --public --default-citation "Budrys, E.,␣
→Budriene., A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/
→data/scales/Trophic specialisation_categorical.tsv"
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --traittype "Specialized on" --provider SCALES_WaspsBeesDatabase --mapping␣
→scientific_name --skip-unmapped --public --default-citation "Budrys, E., Budriene.,␣
→A. and Orlovskyte. S. 2014. Cavity-nesting wasps and bees database." "/data/scales/
→Trophic specialisation_numeric.tsv"
```

3.4.8 IUCN Redlist

IUCN redlist data can be conveniently downloaded using the API. Before you can query the API you need to register for a token. Also if you want to put this data into a public instance you have to make sure to always (automatically) update the data to the latest version in order to comply with the terms of use. For convenience there are some scripts that help with download and update of IUCN data. You have to do some initial preparation and then link additional files into the fennec container:

```
mkdir -p iucn
echo "YOUR IUCN API TOKEN" >iucn/.iucn_token
```

Now edit the `docker-compose.yml` and add to the list of volumes for the `web` service:

```
- "./iucn:/iucn"
```

Then rebuild your web container:

```
docker-compose stop web
docker-compose rm -f web
docker-compose up -d
```

Now you can download and import/update the iucn data in your database with:

```
docker-compose exec web bash -c "cd /iucn;/fennec/util/check_download_update_iucn.sh"
```

This will download the most current version of the IUCN red list via the api and add it to the fennec database. On the first run the traittype is automatically generated. On subsequent runs if the version of IUCN is unchanged nothing happens and if there is a new version the old traits are expired and the new data is loaded. You will notice that only

about half the entries could be mapped by their scientific name. One reason for that is that many species on the red list are species with a small population size endemic to a small geographic region.

> **Warning:** In order to comply with the terms of use of IUCN please add a cron job to your docker host. Unfortunately cron does not work smoothly inside docker but you can try this as well if you feel like it. Otherwise add an entry like this to your host via `crontab -e` (use the correct path):
>
> ```
> 0 * * * * docker-compose -f /path/to/docker-compose.yml exec web bash -c "cd /iucn;/
> →fennec/util/check_download_update_iucn.sh >>iucn_cron.log 2>>iucn_cron.err"
> ```

3.4.9 Bacterial Traits from ProTraits

From the protraits website at http://protraits.irb.hr/ :

> *The ProTraits atlas of prokaryotic traits describes environmental preferences of microbes, interactions with other organisms (including pathogenicity), biochemical phenotypes, resistance to chemicals and other stressors, and utility in industrial applications.*

ProTraits contains 424 phenotypic traits and covers 3,046 bacterial or archeal species. Phenotypes are assigned to microbes using machine learning, using free text available in the scientific literature or the internet. Other sources for trait inference utilized in the ProTraits pipeline are genomic data, see their publication for details:

> "The landscape of microbial phenotypic traits and associated genes", *Maria Brbic, Matija Piskorec, Vedrana Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).*

We use selected traits from the file ProTraits_binaryIntegratedPr0.90.txt. This is a large table with traits as columns and species as rows (with NCBI taxid in column 2). Each cell contains `1`, `0` or `?` denoting a positive label, a negative label, or neither positive nor negative label (at precision >= 0.9) for that organism/trait combination. The columns 3 to 110 contain metabolic traits (we will import them in wide table format). In column 278 there is the trait "pathogenic in mammals" that we will import. Additionally, we will import the traits "bacterial shape", "oxygen requirements", "cell arrangement", and "habitat". Those are split over multiple columns each. E.g. "oxygen requirements" is in columns 275, 276, 277, and 292 (oxygenreq=facultative, oxygenreq=strictaero, oxygenreq=strictanaero, oxygenreq=microaerophilic) We will combine them to a single categorical trait with levels "facultative", "strictaero", "strictanaero", "microaerophilic". For this purpose we use a little perl script available as a gist Data preparation:

```
mkdir -p data/protraits
cd data/protraits
wget http://protraits.irb.hr/data/ProTraits_binaryIntegratedPr0.90.txt
wget https://gist.githubusercontent.com/iimog/0424de0b4efbfe73ef2e9092f8969c06/raw/
→c563320e7ae42c0421c0de6cec14151412c6c4d5/extract_protraits.pl
cut -f2-110 ProTraits_binaryIntegratedPr0.90.txt >protraits_metabolism.tsv
cut -f2,278 ProTraits_binaryIntegratedPr0.90.txt | tail -n+2 | perl -pe 's/$/\t\t\t/'␣
→| grep -v '\?' >pathogenic_in_mammals.tsv
perl extract_protraits.pl ProTraits_binaryIntegratedPr0.90.txt 281 282 283 284 285 |␣
→perl -pe 's/shape=//' >bacterial_shape.tsv
perl extract_protraits.pl ProTraits_binaryIntegratedPr0.90.txt 275 276 277 292 | perl␣
→-pe 's/oxygenreq=//' >oxygenreq.tsv
perl extract_protraits.pl ProTraits_binaryIntegratedPr0.90.txt 218 219 220 221 222␣
→294 | perl -pe 's/cellarrangement=//' >bacterial_cellarrangement.tsv
perl extract_protraits.pl ProTraits_binaryIntegratedPr0.90.txt 234 235 236 237 238␣
→290 295 305 | perl -pe 's/habitat=//' >habitat.tsv
cd -
```

Create the according trait types and import them into fennec:

```
# Create trait types for metabolism
for i in $(cut -f2- data/protraits/protraits_metabolism.tsv | head -n1)
do
        docker-compose exec web /fennec/bin/console app:create-traittype --env prod --
→format categorical_free $i
done
# Create additional trait types
docker-compose exec web /fennec/bin/console app:create-traittype --env prod --format␣
→categorical_free "pathogenic in mammals"
docker-compose exec web /fennec/bin/console app:create-traittype --env prod --format␣
→categorical_free "bacterial shape"
docker-compose exec web /fennec/bin/console app:create-traittype --env prod --format␣
→categorical_free "oxygen requirements"
docker-compose exec web /fennec/bin/console app:create-traittype --env prod --format␣
→categorical_free "bacterial cell arrangement"
docker-compose exec web /fennec/bin/console app:create-traittype --env prod --format␣
→categorical_free "habitat"

# Import metabolism in wide format
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
→prokaryotic traits"\
 --mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
→phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
→Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
→https://doi.org/10.1093/nar/gkw964'\
 --wide-table --skip-unmapped /data/protraits/protraits_metabolism.tsv

# Import the other traits
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
→prokaryotic traits"\
 --mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
→phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
→Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
→https://doi.org/10.1093/nar/gkw964'\
 --traittype "pathogenic in mammals" --skip-unmapped /data/protraits/pathogenic_in_
→mammals.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
→prokaryotic traits"\
 --mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
→phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
→Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
→https://doi.org/10.1093/nar/gkw964'\
 --traittype "bacterial shape" --skip-unmapped /data/protraits/bacterial_shape.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
→prokaryotic traits"\
 --mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
→phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
→Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
→https://doi.org/10.1093/nar/gkw964'\
 --traittype "oxygen requirements" --skip-unmapped /data/protraits/oxygenreq.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
→entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
→prokaryotic traits"\
```

```
--mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
↪phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
↪Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
↪https://doi.org/10.1093/nar/gkw964'\
 --traittype "bacterial cell arrangement" --skip-unmapped /data/protraits/bacterial_
↪cellarrangement.tsv
docker-compose exec web php -d memory_limit=1G /fennec/bin/console app:import-trait-
↪entries --env prod --provider ProTraits --description "The ProTraits atlas of␣
↪prokaryotic traits"\
 --mapping ncbi_taxonomy --public --default-citation '"The landscape of microbial␣
↪phenotypic traits and associated genes", Maria Brbic, Matija Piskorec, Vedrana␣
↪Vidulin, Anita Krisko, Tomislav Smuc, Fran Supek. Nucleic Acids Research (2016).␣
↪https://doi.org/10.1093/nar/gkw964'\
 --traittype "habitat" --skip-unmapped /data/protraits/habitat.tsv
```

> **Warning:** Import of the wide table data massively inflates the trait values stored in the database. Unfortunately most of the values are ? which is not valuable information. In order to avoid importing those uninformative trait values it is planned to add a --ignore-values parameter to the import-trait-entries command. As this is not implemented yet, you can remove those entries manually with these commands:
>
> ```
> docker-compose exec web /fennec/bin/console doctrine:query:sql --connection default_
> ↪data "DELETE FROM trait_categorical_entry WHERE trait_categorical_value_id IN␣
> ↪(SELECT id FROM trait_categorical_value WHERE value='?');"
> docker-compose exec web /fennec/bin/console doctrine:query:sql --connection default_
> ↪data "DELETE FROM trait_categorical_value WHERE value='?';"
> ```

3.5 Multiple data databases

It is possible to have multiple data databases in Fennec. This is useful, both to provide different versions and to provide specific databases for groups of organisms. While projects are always stored in the user database the data database to work on can be selected in the web interface. Users can map their organisms against different data databases (this information is stored independently). However traits mapped to the project are stored without distinguishing database versions.

To create an additional database add to your parameters.yml (for a simpler presentation the irrelevant fractions of the file are not shown, denoted by # ...):

```
parameters:
    # ...
    user_connection: "userdb"
    user_entity_manager: "userdb"
    default_data_connection: "default_data"
    default_data_entity_manager: "default_data"
    versions: 'default_data|alternative_data'
    dbal:
        connections:
            'userdb':
                # ...
            'default_data':
                # ...
            'alternative_data':
                driver: pdo_pgsql
```

```
                host: datadb
                port: 5432
                dbname: fennec_alt_data
                user: fennec_data
                password: fennec_data
                charset: UTF8
    orm:
        auto_generate_proxy_classes: '%kernel.debug%'
        entity_managers:
            'userdb':
                # ...
            'default_data':
                # ...
            'alternative_data':
                connection: 'alternative_data'
                naming_strategy: doctrine.orm.naming_strategy.underscore
                mappings:
                    AppBundle:
                        dir: '%kernel.project_dir%/src/AppBundle/Entity/Data'
                        type: annotation
                        prefix: 'AppBundle\Entity\Data'
```

This adds a new database to the existing `datadb` docker container. You can also add another docker container to the `docker-compose.yml` file and configure the new database in there. In order to initialize the new database execute those commands:

```
docker-compose restart web
docker-compose exec web /fennec/bin/console doctrine:database:create --connection␣
↪alternative_data
docker-compose exec web /fennec/bin/console doctrine:schema:create --em alternative_
↪data
docker-compose exec web /fennec/bin/console doctrine:fixtures:load --em alternative_
↪data -n
```

If your database does not show up in the web interface, double check that you added `alternative_data` to the `versions` in `parameters.yml` and clear the cache as explained above. From now on when you import data and you want it to end up in the `alternative_data` db you have to add `--dbversion alternative_data` to the command. If you do not specify the `--dbversion` option the value from `default_data_entity_manager` in `parameters.yml` will be used.

3.6 Backup

If you followed the setup above all fennec related data is on the host in the `fennec` directory. You should regularly create backup copies of this directory. However, you might want to additionally create dumps from the databases for easy import into other instances. To backup the databases just execute the following commands (repeat for all additional data databases):

```
mkdir -p backup
docker-compose exec userdb pg_dump -U fennec_user --data-only --no-owner fennec_user␣
↪| xz >backup/fennec_user.$(date +%F_%T).sql.xz
docker-compose exec datadb pg_dump -U fennec_data --data-only --no-owner fennec_data␣
↪| xz >backup/fennec_data.$(date +%F_%T).sql.xz
# docker-compose exec datadb pg_dump -U fennec_data --data-only --no-owner fennec_alt_
↪data | xz >backup/fennec_alt_data.$(date +%F_%T).sql.xz
```

3.7 Import database from dump

In order to import a database dump follow this steps (assuming you want to remove all old data before importing). You might want to do this in the `alternative_data` database (see above) instead of `default_data`:

```
docker-compose exec web /fennec/bin/console doctrine:database:drop --force --
→connection default_data
docker-compose exec web /fennec/bin/console doctrine:database:create --connection␣
→default_data
docker-compose exec web /fennec/bin/console doctrine:schema:create --em default_data
# do not load fixtures otherwise there will be unique constraint violations
# replace the backup filename with an existing one
xzcat fennec_default_data.sql.xz | docker-compose exec -T datadb psql -U fennec_data -
→d fennec_data
```

3.8 Upgrade

To upgrade to a new version of FENNEC please review the change log and pay special attention to any breaking changes. Always make a full backup of your database (see above) and all files you modified before upgrading. If there were changes to the database schema special migration steps might be necessary. Double check the change log before you continue. The cleanest way to upgrade (if you are using the docker compose setup) is by replacing the docker container with the latest version like this:

```
# Before you continue: Do the backup as described above!
docker-compose down
docker-compose pull
docker-compose up
```

Thats it. The containers are replaced by the version specified in your `docker-compose.yml` file. So `latest` for the fennec container. You can pin the fennec container to a version or switch to develop by adding the desired label, e.g. `:develop`.

Indices and tables

- genindex
- modindex
- search