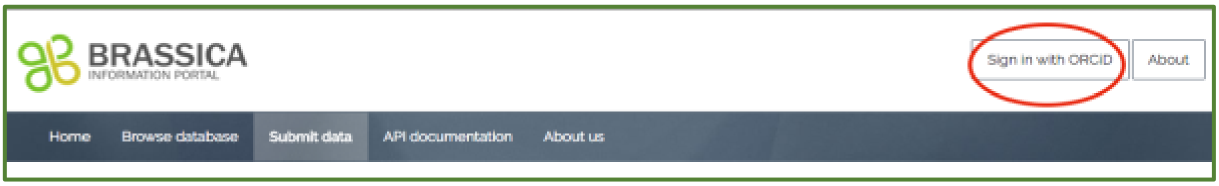
Data submission workshop upload guide

This document is a manual for the steps taken during the data upload workshop on the 07.09.2016. It lists the steps necessary to use the ruby client to submit population data and the wizard to submit trial data to the Brassica Information Portal.
It further names all fields required for the submission of a new experimental plant population and plant trial.

In order to submit content to the Brassica Information Portal, you need to sign in with your ORCiD account. If you don't have an ORCiD account yet, you will be referred to their services from our Sign in.
Please create an account prior to the workshop, as our time is limited.



Your submission is split into two different submissions which need to be executed in the following order:

1) Population Submission (ruby script)
2) Trial Submission (web- interface based)

Population Submission using the ruby client

You submit your experimental plant population first. This is the collection of the genetically different plant material used for your trial. In your trial submission you are asked for the name of your experimental population, which is why it is necessary to submit population information first. The fields required and the population submission client are specific to a diversity foundation set, not a crossing population.

The population submission client is a ruby script parsing information (objects) from a .csv spreadsheet provided by the user to the mapped location in the BIP database via the BIP-API. In order to understand the underlying resource and attribute names used in the script, please see the respective tables in the API-documentation.

Table 1. Fields required* for the submission of an experimental Population. They are presented below by <resource_name>.<attribute_name>.

- Plant_populations.name
- Plant_populations.population_type
- Plant_populations.description
- Plant_population.establishing_organization
- Plant_lines.plant_line_name
- Taxonomy_terms.id
- Plant_varieties.plant_variety_name
- Plant_vatieties.crop_type
- Plant_accessions.plant_accession
- Plant_accessions.year_produced
- Plant_acessions.originating_organisation

*Note that more fields are available for submission of information to the BIP; to find the description of their names, please go to our API-documentation.

During your submission you will be using terminal. Please be familiar with the basics in navigating folders and moving and manipulating files. We will be working in pairs for this part so if you are not comfortable, I will pair you up with a more experienced person.

Install software:

-download ruby 2.3+

-for OS X e.g. use brew:
download brew following the instructions on the website:
http://brew.sh/

then type:

```
brew install ruby
```

-for Windows, use rubyinstaller:
Download the installer by following the instructions on the website:
http://rubyinstaller.org/

Files required:

Download the client and an example input_file.csv here:
https://github.com/TGAC/brassica/tree/master/public/population_submission

Create a .csv file with information on your population, in the fashion as shown in figure 2, which is the input_file.csv you can also retrieve from the link above. This template input_file.csv is reflecting the content of the current commands in the population_submission.rb client and can be altered to your requirements/ available data. Some information on file manipulation is below.

File manipulation:

The population_submission.rb client currently enables the submission of all fields listed in table 1. This section walks you through each step that needs to be taken in order to modify your script and input template for your own population submission The order of steps reflects the logical submission of information rather than the way things appear in the script.

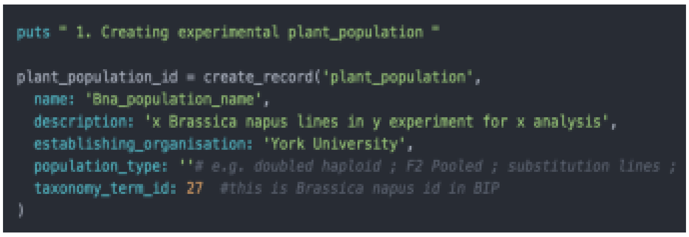
Population description:

For your own population submission, you will have to alter the following information directly in the script:

Beneath " 1. Creating experimental plant_population ", you will see information as displayed in figure 1.
Please enter your population information such as the

- plant_population.name,
- plant_population.description
- plant_population.establishing_organisation
- plant_population.population_type

directly in the script.
Nomenclature examples for the plant_population.name are: Bna_TNDH (Doubled Haploid mapping population derived from cross between Tapidor DH and Ningyou 7) or BnaDFS (a *Brassica napus* diversity foundation set).
If you are going to submit a population with species other than Brassica napus, please contact me to receive the appropriate identifier.



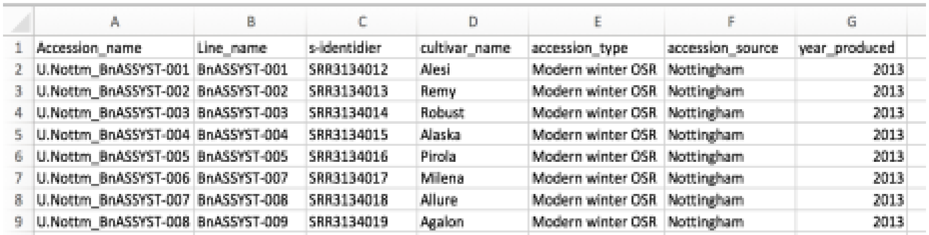
```
puts " 1. Creating experimental plant_population "

plant_population_id = create_record('plant_population',
  name: 'Bna_population_name',
  description: 'x Brassica napus lines in y experiment for x analysis',
  establishing_organisation: 'York University',
  population_type: ''# e.g. doubled haploid ; F2 Pooled ; substitution lines ;
  taxonomy_term_id: 27  #this is Brassica napus id in BIP
)
```

Figure 1 example text describing the plant population in population_submission.rb

The .csv file

With the help of this script you are able to read in the content of your own .csv input file with your population metadata to the Brassica Information Portal.

The template input_file.csv (Figure 2) reflects the current commands in the population_submission.rb client to read in data from that .csv file. It can be altered to the information available on your experimental plant population.

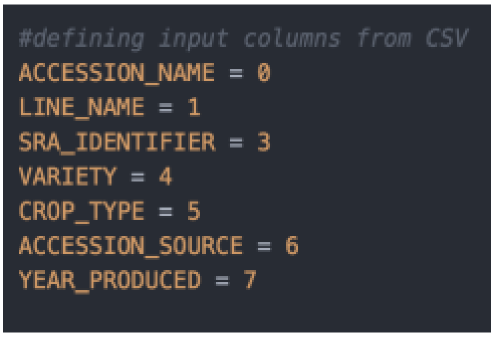


	A	B	C	D	E	F	G
1	Accession_name	Line_name	s-identidier	cultivar_name	accession_type	accession_source	year_produced
2	U.Nottm_BnASSYST-001	BnASSYST-001	SRR3134012	Alesi	Modern winter OSR	Nottingham	2013
3	U.Nottm_BnASSYST-002	BnASSYST-002	SRR3134013	Remy	Modern winter OSR	Nottingham	2013
4	U.Nottm_BnASSYST-003	BnASSYST-003	SRR3134014	Robust	Modern winter OSR	Nottingham	2013
5	U.Nottm_BnASSYST-004	BnASSYST-004	SRR3134015	Alaska	Modern winter OSR	Nottingham	2013
6	U.Nottm_BnASSYST-005	BnASSYST-005	SRR3134016	Pirola	Modern winter OSR	Nottingham	2013
7	U.Nottm_BnASSYST-006	BnASSYST-007	SRR3134017	Milena	Modern winter OSR	Nottingham	2013
8	U.Nottm_BnASSYST-007	BnASSYST-008	SRR3134018	Allure	Modern winter OSR	Nottingham	2013
9	U.Nottm_BnASSYST-008	BnASSYST-009	SRR3134019	Agalon	Modern winter OSR	Nottingham	2013

Figure 2 Input_file.csv is a template that corresponds to the population submission client, available online.

The template file contains the minimal requirements of information for a population submission to the Brassica Information Portal. An addition that is not required is the submission of a list of sequence identifiers from SRA. Make sure you remove the code handling the sequence identifier submission from the script in case you don't have sequence identifiers to submit.

Defining the input to the script



```
#defining input columns from CSV
ACCESSION_NAME = 0
LINE_NAME = 1
SRA_IDENTIFIER = 3
VARIETY = 4
CROP_TYPE = 5
ACCESSION_SOURCE = 6
YEAR_PRODUCED = 7
```

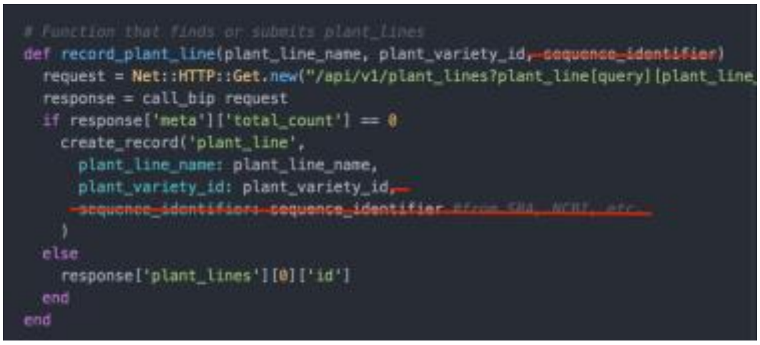
Figure 3 the input variables must correspond to the location of the column headers in your .csv file.

In section "# defining input columns from CSV in the script" you define the script's input parameters using the columns from your .csv.
0 in the script is the first row in your .csv. 1 is the second row in your .csv. For example, in Figure 3, Accession_name information is located in the first column in the .csv template (Figure 2), but needs to be defined as "0" in the script (Figure 3).

Manipulating a function

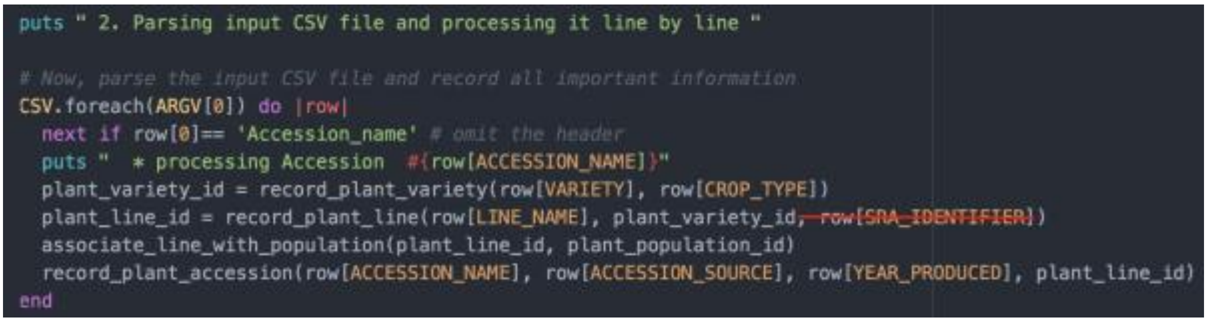
You may not have SRA identifier information for your population. Therefore, you have to remove any code that deals with the submission of this information. The first step is to re-define the input of your script (see Fig.3), where the variable SRA_IDENTIFIER needs to be removed and the input column numbers changed accordingly.

Further, remove sequence_identifier information from the function that defines the submission of plant_line associated information (as shown in Fig. 4). Finally, find the line where the function gets called on the bottom of the script and remove the input variable from there (as shown in Fig 5).



```
# Function that finds or submits plant_lines
def record_plant_line(plant_line_name, plant_variety_id~~, sequence_identifier~~)
  request = Net::HTTP::Get.new("/api/v1/plant_lines?plant_line[query][plant_line_
  response = call_bip request
  if response['meta']['total_count'] == 0
    create_record('plant_line',
      plant_line_name: plant_line_name,
      plant_variety_id: plant_variety_id~~,~~
      ~~sequence_identifier: sequence_identifier #from SRA, NCBI, etc~~
    )
  else
    response['plant_lines'][0]['id']
  end
end
```

Figure 4 function that records plant_line information. Crossed out is the code that would enable sequence_identifier submission.



```
puts " 2. Parsing input CSV file and processing it line by line "

# Now, parse the input CSV file and record all important information
CSV.foreach(ARGV[0]) do |row|
  next if row[0]== 'Accession_name' # omit the header
  puts "  * processing Accession  #{row[ACCESSION_NAME]}"
  plant_variety_id = record_plant_variety(row[VARIETY], row[CROP_TYPE])
  plant_line_id = record_plant_line(row[LINE_NAME], plant_variety_id~~, row[SRA_IDENTIFIER]~~)
  associate_line_with_population(plant_line_id, plant_population_id)
  record_plant_accession(row[ACCESSION_NAME], row[ACCESSION_SOURCE], row[YEAR_PRODUCED], plant_line_id)
end
```

Figure 5 The calling of functions like "record_plant_line" from Fig.4 is located at the end of the script. Crossed out is the code related to SRA sequence identifier submission.

In the same fashion as the ability to submit SRA sequence information has been removed from the script, one can add more objects that facilitate the submission of additional information. The API documentation lists all possible objects that you could submit information to. In the case of Population submission, you can add objects related to the Plant_population, Plant_lines, Plant_accessions and Plant_varieties.

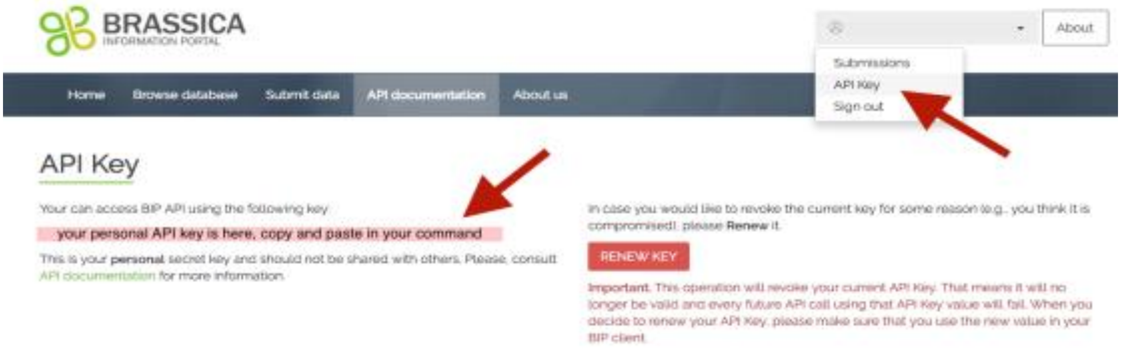
Double-checking the script

- Are all .csv headers correctly associated with the variables in the script?
- Do variables need to be added to the script?
    - If yes: make sure you add them in the appropriate create_record function, with the wording of the fields exactly as shown in the API documentation
- Do variables need to be /deleted from the script?
    - If yes: make sure you delete them in section #defining input columns from CSV, in the create_record function, and at the end of the script, where you call the function and define the input parameters.

API key:

Sign in to BIP to retrieve your API key:

1) Click on API key in the popup bar under your user name.
2) Copy the API key and paste in your command to run the submission client (see below).



API Key

Your can access BIP API using the following key

your personal API key is here, copy and paste in your command

This is your **personal** secret key and should not be shared with others. Please, consult API documentation for more information.

In case you would like to revoke the current key for some reason (e.g., you think it is compromised), please **Renew** it.

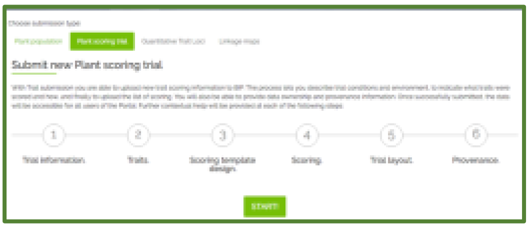
Important. This operation will revoke your current API Key. That means it will no longer be valid and every future API call using that API Key value will fail. When you decide to renew your API Key, please make sure that you use the new value in your BIP client.

Run submission client:

After double-checking your submission file and script, run the following command in your terminal. Make sure your input file and the submission script are located in the same folder or add the paths to their name:

```
ruby population_submission.rb  <your_input_file.csv>  <your_api_key>
```

Trial Submission using the wizard



Submit new Plant scoring trial

After submitting your Population, you can submit the trait scoring data (your measured traits) of your trial. This is a 6-step process, during which you also submit metadata that describes your trial.
The wizard walks you through all the steps, offering compulsory and optional fields to be filled out. On the right side, you see a list of all fields during the submission process. Those fields marked with * are compulsory for the submission.
Step 1 asks you for general trial information. In this step you link your previously submitted plant_population to your plant_trial, by selecting your

Trial submission

Step 1 - Trial information

Plant trial name*

Project name*

Experimental plant population*

Trial year*

Trial description*

select: data status: raw vs. processed (analysed) data

Institute*

Terrain

Soil type

Statistical factors

Country*

Place name*

Trial location site name

Latitude

Longitude

Altitude

experimental plant_population from a drop-down menu.

In step 2 you define the traits which you have investigated during your trial. Some fields are compulsory, following the Crop Ontology model of <Trait><Method><Scale>. Other information can be added in case it is available. We advise to add as much information as possible to make your data reusable and comparable.

In step 3 you specify what information you will upload together with your trait scores. This can be the number of technical replicates or the trial design in case you want to submit raw data. Also, you specify whether you give information on the plant_lines or plant_varieties associated with your germplasm.

In step 4 you will be able to download the template you use for the submission of your trait data. It has been created according to the choices you have made so far in your submission (e.g. traits to be submitted). If you see that the headers don't correspond to the data you want to submit, navigate back to previous submission steps and amend them accordingly. An altered template can then be downloaded at this step. Please **be careful when pasting your trait scores** beneath the correct headers, as they may not appear in the order they are recorded in your source spreadsheet.

Step 5 is optional, where you can submit an image of the trial layout, in case you submit raw data. Such an image would be helpful to interpret patterns in the raw data.

Step 6 asks you to fill in some information about the provenance of the data you just submitted. You can choose to put an embargo on the data and wait with the submission until for

Step 2 -Traits

Select trait descriptors*

When adding new trait descriptor:

Descriptor name*	Materials	Precautions
Trait category*	Instrumentation required	Scoring method
Units of measurements*	Calibrated against	Possible interactions
Score type	Likely ambiguities	Additional annotations
Where to score	Controls	

Step 3 - Scoring template design

Select genetic material origin

plant lines

plant varieties

specify technical replicate numbers (raw data)

select design factors: block plot rep etc (raw data)

Step 4 - Scoring Template Submission

download .csv file template

add trait scores to corresponding header

add plant scoring unit (=sample_id)*, plant accession*, originating organisation*,

Plant line or variety*

upload .csv file*

Step 5 Submit Trial Layout

Submit image of trial layout

Step 6- Provenance

Data owned by

Data provenance

Comments

Visibility-public/private*

example your paper is ready to be published.	