Command Line

Table of Contents:

Overview

BioNetDB implements the bionetdb.sh command line with the following syntaxis:

```
bionetdb.sh [-h|--help] [--version] <command> [options]
```

General usage documentation

In this section you will learn how to use the command lines, some examples are provided using *bionetdb.sh* command line.

Basic command line conventions

BioNetDB follows the most standard conventions when implementing command lines, the most relevant ones are:

- **single-character** parameters start with only one *hyphen* symbol and can be either lower-case or upper-case, e.g. *-h* to print the help or *-S* to provide session id; while **multi-character** parameters start with two *hyphens* and are always lower-case, e.g. *--help* to print the help or *--name* to provide a name.
- in the *Usage* the optional parameters are written always between ***[** square brackets **]***, in the following example *help, version* and *option* are not mandatory:

  ```
  ./opencga.sh [-h|--help] [--version] <command> [options]
  ```

- in the *Usage* the mandatory parameters are written between symbols **<** and **>**, like <*command*> in the previous example.

Executing the top level command line

You can execute the command line without any argument to get the usage help (you can also provide *-h* parameter):

```
Program:     BioNetDB (OpenCB)
Version:     1.0.0
Description: BioNetDB implements a storage engine to work with biological
networks using Neo4j, a NoQSL Graph database

Usage:       bionetdb.sh [-h|--help] [--version] <command> [options]

Commands:
              import  Import the built data models in format CSV files
into the database
               query  Query and fetch data from BioNetDB database using
this command line
```

Executing a specific *command*

When one *command* is executed without any other argument a specific help is shown, this specific help shows a brief description and the different options available with a brief description, as you can see in the following example:

```
Usage:   bionetdb.sh import [options]

Options:
      --create-csv-files              Create the CSV files from the input
biological files [false]
create-csv-files parameter, it contains the biological files to convert to
CSV files) [null]
    * -i, --input          STRING     Input directory where the CSV files
are located (when used with --
      -o, --output         STRING     Output directory where to save the
CSV files to import (used with the -      -create-csv-files parameter)
[null]
      -L, --log-level      STRING     Set the logging level, accepted
values are: debug, info, warn, error and fatal [info]
      -C, --conf           STRING     BioNetDB configuration.json file.
[null]
      -h, --help                      Display this help and exit
```

Common options

Some of these options are available for most of the *commands*, these are:

- -C, --conf: If not defined, the command line will assume that the configuration folder can be found in the parent folder where the executable is found (../conf). If that is not the case or if the user wants to use other configuration files available in a different path, the path should be provided.
- -h, --help: Shows the help with the list of options. Generally this is not necessary because there is almost always a mandatory parameter to be provided, although it is needed in a few cases if user wants to see the options.
- --log-file: By default, all the logs generated by the command lines are printed in the screen. However, the user might want to redirect the logs to another file. For those cases, the user will need to provide the file where the logs will be stored.
- -L, --log-level: There are 5 different log levels: 'error', 'warn', 'info', 'debug', 'trace'