# OBITools Documentation

***Release 0.1.3***

**Eric Coissac**

December 10, 2009

# CONTENTS

Contents:

CHAPTER

ONE

# OBITOOLS SCRIPTS

OBITools scripts are developed mainly for manipulating large sequence files generated by the next generation sequencers.

Contents:

## 1.1 File formats usable with OBITools

### 1.1.1 The sequence files

Sequences can be stored following various format. OBITools knows some of them. The central format for sequence files manipulated by OBITools scripts is the *fasta format*. OBITools extends the fasta format by specifying a syntax to include in the definition line data qualifying the sequence. All file formats use the *IUPAC* code for encoding nucleotides and amino-acids.

#### The IUPAC code

The International Union of Pure and Applied Chemistry (IUPAC) defined the standard code for representing protein or DNA sequences.

## Nucleic IUPAC Code

| Code | Nucleotide |
|---|---|
| A | Adenine |
| C | Cytosine |
| G | Guanine |
| T | Thymine |
| U | Uracil |
| R | Purine (A or G) |
| Y | Pyrimidine (C, T, or U) |
| M | C or A |
| K | T, U, or G |
| W | T, U, or A |
| S | C or G |
| B | C, T, U, or G (not A) |
| D | A, T, U, or G (not C) |
| H | A, T, U, or C (not G) |
| V | A, C, or G (not T, not U) |
| N | Any base (A, C, G, T, or U) |

## Peptidic one and three letters IUPAC code

| 1-letter | 3-letters | Amino acid |
|---|---|---|
| A | Ala | Alanine |
| R | Arg | Arginine |
| N | Asn | Asparagine |
| D | Asp | Aspartic acid |
| C | Cys | Cysteine |
| Q | Gln | Glutamine |
| E | Glu | Glutamic acid |
| G | Gly | Glycine |
| H | His | Histidine |
| I | Ile | Isoleucine |
| L | Leu | Leucine |
| K | Lys | Lysine |
| M | Met | Methionine |
| F | Phe | Phenylalanine |
| P | Pro | Proline |
| S | Ser | Serine |
| T | Thr | Threonine |
| W | Trp | Tryptophan |
| Y | Tyr | Tyrosine |
| V | Val | Valine |
| B | Asx | Aspartic acid or Asparagine |
| Z | Glx | Glutamine or Glutamic acid |
| X | Xaa | Any amino acid |

### The fasta format

The fasta format is certainly the most widely used sequence file format. This is certainly due to its great simplicity. It was originally created for the ‘’Lipman” and ‘’Pearson” FASTA program. OBITools use in more of *the classical fasta*

*format* several extended version of this format where structured data are included in the title line.

## The extended OBITools fasta format

The *extended OBITools Fasta format* is a strict *fasta format file*. The file in *extended OBITools Fasta format* can be readed by all programs reading fasta files.

Difference between standard and extended fasta is just the structure of the title line. For OBITools title line is divided in three parts :

- Seqid : the sequence identifier
- key=value; : a set of key/value keys
- the sequence definition

```
>my_sequence taxid=3456; direct=True; sample=A354; this is my pretty sequence
ACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGT
GTGCTGACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTGTTT
AACGACGTTGCAGTACGTTGCAGT
```

Following these rules, the title line can be parsed :

- The sequence identifier of this sequence is *my_sequence*
- **Three keys are assigned to this sequence :**
  - Key *taxid* with value *3456*
  - Key *direct* with value *True*
  - Key *sample* with value *A354*
- The definition of this sequence is this is *my pretty sequence*

Key value can be any valid python expression. If a key value cannot be evaluated as a python expression, it is them assumed as a simple string. Following this rule, taxid value is considered as an integer value, direct value as a boolean and sample value is not a valid python expression so it is considered as a string value.

## The classical fasta format

In fasta format a sequence is represented by a title line beginning with a **>** character and the sequences by itself following *The IUPAC code*. The sequence is usually split other severals lines of the same length (expected for the last one)

```
>my_sequence this is my pretty sequence
ACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGT
GTGCTGACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTGTTT
AACGACGTTGCAGTACGTTGCAGT
```

This is no special format for the title line excepting that this line should be unique. Usually the first word following the **>** character is considered as the sequence identifier. The end of the title line corresponding to a description of the sequence.

Several sequences can be concatenated in a same file. The description of the next sequence is just pasted at the end of the description of the previous one

```
>sequence_A this is my first pretty sequence
ACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGT
GTGCTGACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTGTTT
AACGACGTTGCAGTACGTTGCAGT
>sequence_B this is my second pretty sequence
ACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGT
GTGCTGACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTGTTT
AACGACGTTGCAGTACGTTGCAGT
>sequence_C this is my third pretty sequence
ACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGT
GTGCTGACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTACGTTGCAGTGTTT
AACGACGTTGCAGTACGTTGCAGT
```

#### The genbank sequence format

#### The EMBL sequence format

### 1.1.2 The taxonomy files

Many OBITools are able to take into account taxonomic data. These data are manipulated following the NCBI taxonomy.

#### The NCBI taxonomy dump files

#### The OBITools formated taxonomy

### 1.1.3 The ecoPCR files

ecoPCR is a software developed in LECA. It simulates a PCR experiment by selecting in a sequence database, sequences matching simultaneously two primers sequences in a way allowing a PCR amplification of a DNA region.

### 1.1.4 The ecoPrimer files

### 1.1.5 The OBITools files

## 1.2 File format conversions

Several OBITools exist for converting files from one format to another. As *fasta file* is the central format for OBITools, many of these converters convert to *extended OBITools fasta format*.

### 1.2.1 Convert to extended OBITools fasta format

#### Convert sequence files to extended OBITools fasta format

**convert2fasta.py** [options] [filename 1] [filename 2] ...

Convert sequence files to the extended OBITools fasta format. If no file name are specified data are read from standard input.

## obitools common options

**-h, -help**
show this help message and exit

**-DEBUG**
Set logging in debug mode

**-no-psyco**
Don't use psyco even if it installed

## convert2fasta.py specific options

**-genbank**
input file is in *genbank format*

**-embl**
input file is in *embl format*

**-fna**
input file is in fasta nucleic format produced by 454 sequencer pipeline

**-nuc**
input file contains nucleic sequences

**-prot**
input file contains protein sequences

## example

for converting a genbank file to fasta

```
% convert2fasta.py --genbank --nuc sequences.gb > sequences.fasta
```

### Convert ecoPCR result files to extended OBITools fasta file

## 1.2.2 Convert taxonomic data

### Convert NCBI taxdump to binary formated OBITools taxonomy database

**buildOBITaxonomy.py** -t <taxdump dir> -d <db name>

Convert an text dump directory of the NCBI Taxonomy database to the binary format used by ecoPCR and many *OBITools* scripts. An archive corresponding to this directory can be downloaded at the following URL

ftp://ftp.ncbi.nlm.nih.gov/pub/taxonomy/

## obitools common options

**-h, -help**
show this help message and exit

**-DEBUG**
Set logging in debug mode

**-no-psyco**
Don't use psyco even if it installed

### taxonomy related options

**-d <FILENAME>, -database=<FILENAME>**
ecoPCR taxonomy Database name

**-t <FILENAME>, -taxonomy-dump=<FILENAME>**
NCBI Taxonomy dump repository name

### example

for building a new taxonomy database named *ncbitaxonomy* from a taxdump dir

```
% curl ftp://ftp.ncbi.nlm.nih.gov/pub/taxonomy/taxdump.tar.gz | tar zxf -
% buildOBITaxonomy.py --taxonomy-dump taxdump --database ncbitaxonomy
```

## 1.2.3 Convert to tabular data files

#### Convert extended OBITools fasta file to a tabular format

CHAPTER

TWO

# INDICES AND TABLES

- *Index*
- *Module Index*
- *Search Page*

# INDEX

## Symbols



## C



## O



## T