

# User's guide: Manual for Metaxa 1.0

This is a guide to install and use the software utility Metaxa. The software is written for Unix-like platforms, and should work on nearly all Linux-based systems, as well as MacOS X.

## Contents of this manual

1. Detailed installation instructions
2. Usage and commands
3. Output files
4. Algorithm and implementation
5. Running Metaxa's analysis steps separately
6. 'Undocumented' features
7. License information

## 1. Detailed installation instructions

The README.txt file bundled with the script provides a quick installation guide.

In order to install certain packages, you might need to have superuser privileges. For installation on Mac, you will have to install the Apple Xcode package available on your MacOS X System DVD in order to be able to compile programs. Please talk to your system administrator if you feel unsure about these steps. Note that the packages are mandatory and that you should not proceed unless these criteria are fulfilled.

[If you don't have superuser privileges on your machine: Create a directory within your user directory, e.g. /home/user/bin/, and to store all required binaries there. By adding this directory to your PATH, any software placed in the directory will behave as if installed for all users using superuser privileges. If you use the bash shell, you can add a bin directory to your PATH, by adding the line "export PATH=\$PATH:\$HOME/bin/;" to the file .profile in your home directory. The process of adding items to one's PATH varies among systems and shells. Close the terminal and open a new one for this change to take effect.]

Perl needs to be installed on the computer. Most Unix-based systems including Linux and MacOS X have Perl pre-installed. You can check this by opening a command line terminal and type "perl -v". In case Perl is not installed you have to download (<http://www.perl.org>) and compile the program.

Download and install HMMER version 3 (<http://hmmmer.janelia.org/software>). The current version of Metaxa relies on HMMER version 3. Metaxa will **not** work with earlier versions of HMMER. Download the HMMER package source code to your preferred directory such as /home/user/. Open a command line terminal, move into the directory with "cd /home/user/" and unpack the tarball with "tar xvfz hmmer-3.0.tar.gz". Now, you must compile HMMER from source files. To compile it from source, enter the new directory and follow the installation instructions in the file INSTALL.

If you have trouble compiling HMMER, you can try to use the pre-compiled binaries available at the HMMER home page. After download, and unpacking of the tarball, the binaries are located in the binaries directory contained within the newly created HMMER directory. Move into the binaries directory and move all of its contained files into your preferred bin directory (usually either /usr/local/bin/ or your own bin directory,

/home/user/bin/). The HMMER package should now be installed on your computer; you can check this by typing “hmmsearch -h” in the terminal and press enter; you should now see HMMER output.

Download and install the BLAST package (<ftp://ftp.ncbi.nlm.nih.gov/blast/executables/release/LATEST/>) for sequence similarity searches. The current version of Metaxa relies on BLAST, **not** BLAST+, and was written with version 2.2.24 in mind. It should work with any 2.2 version of BLAST. Download the BLAST package for your operating system to your preferred directory. Open a command line terminal, move into the directory with “cd /home/user/” and unpack the tarball with “tar xvfz blast-2.2.24-platform.tar.gz”. Move into the bin directory inside the newly created BLAST directory, and move all of its contained files into your preferred bin directory. Alternatively, you can add the BLAST bin directory to your PATH. The BLAST package should now be installed on your computer; you can check this by typing “blastall” in the terminal and press enter; you should now see the listing of BLAST options.

Download and install the MAFFT (<http://mafft.cbrc.jp/alignment/software/>) for multiple alignment. The current version of Metaxa relies on MAFFT version 6. MAFFT is not critical for Metaxa’s core functions, but is used for automatically creating alignments of uncertain sequences. Instructions for installing MAFFT are available on the MAFFT download page.

Go to <http://microbiology.se/software/metaxa> in order to download the Metaxa package. Download it to your preferred directory. Unpack the downloaded tarball with “tar xvfz metaxa.tar.gz”. A directory called Metaxa will be created. You will see the following files and directories inside it: metaxa, metaxa\_x, metaxa\_c, install\_metaxa, the metaxa\_db directory (containing the Hidden Markov Models and a BLAST database), the user’s guide, the README.txt file, the license.txt file as well as test input files. Enter the directory, and type “./install\_metaxa”. Press enter and follow the on-screen instructions. You will be prompted for whether you have superuser privileges, and where you want Metaxa to be installed. If Metaxa is successfully installed you should see its help message when typing the command “metaxa --help”.

## 2. Usage and commands

**For the *very* impatient only:** follow the brief installation instructions in the file README.txt. To check for SSU rRNA sequences in the file test.fasta, you would then type “metaxa -i test.fasta -o test” on the command line.

**For all other users:** Metaxa accepts input in the FASTA format. As it pre-processes the input sequences it is possible to input both aligned and unaligned FASTA files, containing both DNA and RNA sequences. By default, Metaxa outputs ten files; one summary file of the entire run, one more detailed table of results, one “graphical” representation of hits, one FASTA file of all identified SSU sequences, and one FASTA file for each of the six possible origins. To list all the available options for Metaxa, type “metaxa --help”. You can use the test.fasta file that comes bundled with the software for a test run. This file contains 50 randomly selected SSU entries, ten of each origin, as well as 10 non-SSU sequences. In the simplest case, Metaxa is run by “metaxa -i input\_file -o output”. Below is a listing of all options Metaxa accepts. Boolean options can be turned on with “T”, “true” or “1” and off using “F”, “false” or “0”.

### Main options:

-i {file}	Nucleotide FASTA input file to investigate. Metaxa accepts both aligned and unaligned FASTA. If no input is specified, Metaxa will read sequences from standard input, which means that FASTA sequence can be piped into Metaxa.
-o {file}	Base for the file names of the output files. Suffixes will be added automatically. Defaults to metaxa_out.
-p {directory}	A path to a directory containing HMM-profile collections representing SSU rRNA conserved regions. By default, Metaxa assumes to find the databases in the metaxa_db directory, located in the same directory as Metaxa itself.
-d {database}	The BLAST database used for classification. By default, Metaxa assumes to find the databases in the metaxa_db directory, located in the same directory as Metaxa itself.
--date {T or F}	Adds a date and time stamp to the output file. This can be useful e.g. if Metaxa is part of a pipeline where input files with the same name could cause overwriting of important data. Off (F) by default.

### Sequence selection options:

-t {b, bacteria, a, archaea, e, eukaryota, m, mitochondrial, c, chloroplast, A, all}	Set of profiles to use for the search (comma-separated). Accepts any list of sets, e.g. "bacteria,chloroplast", "m,c" or "eukaryota". Can be used to restrict the search to only a few SSU types to save time, if one or more of the origins are not relevant to the dataset under study. Default is to use all (the "all" option).
-E {value}	Domain E-value cutoff a sequence must obtain in the HMMER-based step to be included in the output. Default = 1e-5.
-S {value}	Domain score cutoff for a sequence must obtain in the HMMER-based step to be included in the output. Default = 0.
-N {value}	The minimum number of domains that must match a sequence for it to be included in the output. Setting the value lower than two will increase the number of false positives, while increasing it above two will decrease Metaxa's detection abilities on fragmentary data. Default = 2.
-M {value}	Number of top BLAST matches that should be considered in classification. Default = 5.
-H {value}	The number of points that the predicted origin of the Metaxa Extractor is given. Default is the same as the number of sequences used for classification (-M option above), which is set to 5 by default.
--selection_priority {sum, domains, eval, score}	<p>Determines what will be of highest priority when assessing the origin of the sequence. Options are:</p> <ul style="list-style-type: none"><li>- sum, which sums the scores for each profile match and divides the sum by the number of profiles of the given type</li><li>- domains, which uses the number of domains retrieved of a given type</li><li>- eval, which uses the average E-value of the found hits</li><li>- score, which uses the average score of the found hits</li></ul> <p>Default is to use sum (sum of scores).</p>
--search_eval {value}	The actual E-value cutoff used in the HMMER search. High numbers may slow down the process. Should never be set to a lower value than the -E option. Cannot be used in combination with the --search_score option.

	Default is 0.01.
--search_score {value}	The score cutoff used in the HMMER search. Low numbers may slow down the process. Should never be set to a higher number than the -S option. Cannot be used in combination with the --search_eval option. Default is to use E-value cutoff (see --search_eval above), not score.
--blast_eval {value}	The E-value cutoff used in the BLAST search. High numbers may slow down the process. Cannot be used in combination with the --blast_score option. Default is 1e-15.
--blast_score {value}	The score cutoff used in the BLAST search. Low numbers may slow down the process. Cannot be used in combination with the --blast_eval option. Default is to use E-value cutoff (see --blast_eval above), not score.
--blast_wordsize {value}	The word-size used for the BLAST-based classification. Lower numbers will slow down the process significantly, while higher numbers may potentially decrease classification accuracy. Default is 14.
--allow_single_domain {e-value,score or F}	Allow inclusion of sequences that only find a single domain, given that they meet the more stringent E-value and score thresholds specified. By default, single domains are allowed, with E-value cutoff 1e-10 and score cutoff 0 ("1e-10,0").
--allow_reorder {T or F}	Allows profiles not to be in the expected order (1-9) on the extracted sequences. If turned off, a file of potential chimeric sequences (with profile matches in the wrong order) is written, allowing for rudimentary chimera detection. This can be used on full-length sequences. On fragmented sequences, however, there is a risk of missing true positives increases if this option is turned off. On (T) by default.
--complement {T or F}	If on, Metaxa checks both DNA strands for matches to HMM-profiles. On (T) by default.
--cpu {value}	The number of CPU threads to use. Metaxa performs significantly faster using more CPUs. Default is 1.
--multi_thread {T or F}	Multi-thread the HMMER-search. On (T) by default if the number of CPUs is larger than one (--cpu option > 1), else off (F).
--megablast {T or F}	Uses megablast for classification for better speed but less accuracy. Off (F) by default.

### Output options:

--summary {T or F}	If on, Metaxa outputs a summary of results. File suffix is ".summary.txt". On (T) by default.
--graphical {T or F}	If on, Metaxa outputs "graphical" text representations of where in each sequence the conserved domains were found. File suffix is ".graph". On (T) by default.
--fasta {T or F}	If on, FASTA-formatted files containing the extracted SSU sequences are written. One file for each origin is written, plus an extraction file containing all SSUs identified in the first analysis step. On (T) by default.
--table {T or F}	If on, Metaxa saves table format output of probable SSU sequences, separately for each origin. Off (F) by default.
--not_found {T or F}	If on, Metaxa outputs a list of entries that do <i>not</i> seem to be SSU sequences. File suffix is "_not_found.txt". Off (F) by default.
--align {a, all, u,	Outputs alignments of BLAST matches to each query in all (a), uncertain (u) or no (n) cases. Requires MAFFT to be installed. Default is to output

uncertain, n, none}	alignments in uncertain cases (u).
--truncate {T or F}	Removes ends of SSU sequences if they are outside of the SSU region. If off, the whole input sequence is saved. On (T) by default.
--guess_species {T or F}	Writes a species guess based on the BLAST-matches to the FASTA definition line. This guess can be pretty far off. Off (F) by default.
--silent {T or F}	Suppresses printing of progress info to screen. Off (F) by default.
--graph_scale {value}	Sets the scale of the graphical output. If the provided value is zero, a percentage view is shown. Default is 0.
--save_raw {T or F}	Saves all raw data for searches etc. instead of removing it when finished. Saves data to a directory with the suffix "_metaxa_raw_output". Off (F) by default.

#### Information options:

-h	Displays the help message.
--help	Displays the help message.
--bugs	Displays the bug fixes and known bugs in this version of Metaxa.
--license	Displays licensing information.

### 3. Output files

Metaxa outputs a number of files, depending on what is selected by the user (see Usage and Commands above). By default, seven FASTA-files, a table of extraction results, a file containing graphical representation of putative SSU sequences, and a summary file is written. In addition, tables of putative SSU sequences, lists of non-SSU entries, and sequence alignments can be written on request by the user. There is also an option to preserve all the intermediate data generated by the HMMER and BLAST searches.

#### FASTA-output

Metaxa generates one FASTA file for each origin (archaea, bacteria, eukaryota, chloroplast, and mitochondria), one file containing sequences of uncertain origin, and one file with all SSU sequences identified and extracted in the first step. Sequences in these files are marked according to their origin. Sequences whose origin Metaxa could not establish with certainty, but for which enough data were available to allow a qualified guess as to the origin of the sequences, are marked with a “#” character at the end of the definition line. A certain sequence may look like this:

```
>gi|117927211 Bacterial 16S SSU rRNA
GTTTGATCCTGGCTCAGGACGAACGCTGGCGGCGTGCTTAACACATGCAAGTCGAGCGGA...
```

Note that Metaxa has added the type of the SSU sequence (“Bacterial 16S SSU rRNA”) to the definition line in the example above. An uncertain sequence could look like this:

```
>AABL01000014.4508.5931 Putative Chloroplast 16S SSU rRNA #
GAACGCTAGAAATATACATTACACATGCAAATTTATGATAATATCATAGTGAATAGGTGA...
```

The extraction file contains all sequences identified as SSUs by metaxa\_x (the first step of the analysis). The sequence entries in that file contain information on what domains that were found and what origin that is most likely based on the profile search. An example is shown below:

```
>A16379.1.1496|B Predicted Bacterial 16S SSU rRNA (1447 bp) From
domain V11 to V9r on main strand Found domains: V11 V21 V2r V31 V3r
V41 V4r V51 V5r V61 V71 V81 V8r V91 V9r
CAGGCTTAACACATGCAAGTCGAACGGTAGCACGAAGGACTTGCTCCTTGGGTGACGAGT...
```

## Summary

A summary of the Metaxa run is written to a file with the suffix “.summary.txt”. In this file the statistics of the run is collected, as are the starting and ending times for the run. Also, lists of the identifiers of extracted SSU sequences are written to this file, one list for each origin. The first section of the file shows the data from the extraction step. The second section is associated with the second classification step. After the second section, the lists of entries of different origins are found. An example of parts of a summary file is shown below:

Metaxa run started at Mon Mar 14 10:07:52 2011.

```
-----
Number of sequences in input file:          100
Sequences detected as SSU rRNA by Metaxa: 100
  On main strand:          91
  On complementary strand:    9
SSU sequences by preliminary origin:
  Archaea:          0
  Bacteria:         0
  Eukaryota:        0
  Chloroplast:     100
  Mitochondria:    0
  Other:           0
-----
```

```
-----
Number of SSU rRNA sequences to be classified by Metaxa:    100
Number of SSU rRNA having at least one database match:      100
Number of SSU rRNA successfully classified by Metaxa:       100
Number of uncertain classifications of SSU rRNA sequences:  0
Total number of classifications made by Metaxa:             100
Number of SSU rRNA sequences assigned to each origin:
  Archaea:          0
  Bacteria:         0
  Eukaryota:        0
  Chloroplast:     100
  Mitochondria:    0
  Uncertain:       0
-----
```

Sequences of archaeal origin (16S):

Sequences of bacterial origin (16S):

Sequences of eukaryote origin (18S):

Sequences of chloroplast origin (16S):

```
Acorus_americanus_AcamCr001
Aethionema_cordifolium_AecoCr001
...
Welwitschia_mirabilis_WemiC_r001
Zea_mays_ZemaCr113
-----
```

Sequences of mitochondrial origin (12S and 16S):

Sequences of uncertain origin:

Metaxa run finished at Mon Mar 14 10:08:42 2011.

## Graphical representations

Metaxa writes graphical (ASCII) representations of where in each sequence the various conserved regions were found to a text file with the suffix “.graph”. Separate graphs are written for each origin and strand, which means that each sequence entry may be present more than once in this file, if it have matches to HMM-profiles from more than one origin. This makes it possible to manually inspect how Metaxa has evaluated each sequence. The graphical representations look like this:

```
B matches on main strand:
>> id|454_30|gi|50402825|gb|AY687385.1|    403 bp
-----V5l=====V5r=====
*****
```

The first row shows the type of the entries below, as well as the strand they are found on. Each entry begins with the characters “>>”, followed by the sequence identifier, and its length. Below the identifier row, the sequence graph is shown. By default, all sequences are scaled so that they are of equal length, and the domains are placed according to their *relative* position in the sequence. The characters that are used in the graphical representations are explained in the table below.

Feature	Description
-	Part of the sequence without any conserved domain (variable region).
V1l	Start of a conserved domain.
=	Continuation of a conserved domain.
>	Indicates that one conserved domain goes into the next. Domains are normally not overlapping, so this could be an indication of a bad input sequence.

The line of asterisks indicates the end of one set of matches. Note that the graph should be viewed with a non-proportional font, such as Courier, if loaded into, e.g., Word.

## Extraction results table

The full results of the Metaxa extraction is saved to a file with the suffix “.extraction.results”. This file consists of tab-separated columns containing various information on each SSU sequence found. The file can be easily imported into programs such as Excel. The contents of the columns (from left to right) are explained in this table:

Column	Description
ID	The identifier of the query sequence.
Length	The length of the query sequence.
Origin	A one-letter abbreviation of the sequence origin. A = archaeal, B = bacterial, C = chloroplast, E = eukaryote, M = mitochondrial 16S, N = mitochondrial 12S.
Strand	A zero (0) if the SSU was found on the main strand, a one (1) if it was found on the complementary strand.
Domains	The number of conserved domains for the most likely origin that was found in the sequence.
Average E-value	The average E-value for these domains.

Average score	The average score for these domains.
Start	The starting position of the first domain.
End	The ending position of the last domain.
First domain	The domain that is located first on the sequence.
Last domain	The domain that is located last on the sequence.
Chimera	The word "Chimeric" if the sequence was marked as a potential chimera. Empty if not. Sequences will only be marked as chimeric if the <code>--allow_reorder</code> option is turned off. Note that this is <i>not</i> a robust measure against chimeras of all kinds.
Specific origin information	A collection of information of <i>all</i> possible origins for the given query. Each entry is a space-separated list, containing the origin type, the number of domains of that type, the average E-value, and the average score, e.g. "N: 4 8.2e-11 43.475"

### Extraction results table

If table output is turned on, Metaxa will save statistics of every profile set that the sequence in question matches to in a file with the suffix ".hmmer.table". This file consists of tab-separated columns containing information on the SSU sequence found. The contents of the columns (from left to right) are explained in this table:

Column	Description
ID	The identifier of the query sequence.
Length	The length of the query sequence.
List of hits	Each new column contains information of a profile match. Each column is organised as follows: "<starting position> - <ending position>: <name of matching profile> (<score>, <E-value>)".

As in the graphical output file, the table file is divided into sections. Each section represents one group of sequences, and begins with the line "X matches on main strand:", and ends with a line of asterisks.

### List of non-SSU sequences

If not-found output is turned on, Metaxa will write a list of sequences for which no conserved SSU regions could be found to a file with the suffix "\_not\_found.txt". The file contains only the identifiers of the non-SSU sequences.

### Sequence alignments

By default, Metaxa saves alignments of sequences of uncertain origin to a directory with the suffix "\_alignments". The user may specify to instead align all SSU sequences by using the "--align all" option (note that this would increase the runtime significantly). The five best BLAST matches are aligned to the query sequence, and saved to an aligned FASTA file with the name "<query identifier>.aligned.fasta".



## Chimeric sequences

If the option `--allow_reorder` is turned off, Metaxa will save an additional FASTA file containing sequences that are suspected to be chimeric. These are sequences with domains located in the wrong order. This is useful on full-length or near full-length data sets, but should not be used on short reads as it could increase the number of false negatives when run on short sequences.

## Raw data

If the option to save all raw data is turned on, Metaxa will save all data from the pre-processing, HMMER-search, BLAST-search, as well as a file of raw statistics into a directory with the suffix `“_metaxa_raw_output”`.

## 4. Algorithm and implementation

The main design goal for Metaxa is to achieve fast and accurate extraction of SSU sequences in large data sets, without introducing a large number of false positives. To be able to reach a high speed, Metaxa relies on the HMMER3 software, which allows for extremely fast comparisons of HMM-profiles to a sequence set. HMMER is used to extract a subset of the input sequences that is subsequently analysed for origin. Thus, the program does not have to consider a large number of non-SSU sequences that would slow down the classification process.

To achieve high detection accuracy, Metaxa uses multiple HMM-profiles representing conserved domains in the SSU sequence. In addition, separate sets of HMM-profiles for SSU sequences of archaeal, bacterial, eukaryal, mitochondrial and chloroplast origin are utilised. To accommodate for the fact that there are both 16S and 12S mitochondrial SSU sequences (the latter chiefly found in animals), two distinct sets of profiles are used to accurately detect both of these categories. The archaeal, bacterial, and eukaryote profile sets are taken from the V-Xtractor 2.0 software, while the chloroplast and mitochondrial 12S and 16S sets are newly generated following the same procedure as for the V-Xtractor profiles. To avoid false positive matches, Metaxa by default requires at least two such conserved domains to be found on a query sequence. This criterion brings down the false positive rate to about 0.0001%.

As several of the conserved domains are closely similar in e.g. bacteria and chloroplasts, the initial classification made by the HMMER-based step will not be perfect. Thus, the results of the extraction are sent to a BLAST-based classification step, where each SSU sequence is matched to a manually inspected database of archaeal 16S, bacterial 16S, eukaryal 18S, chloroplast 16S and mitochondrial 12S and 16S sequences. Each possible origin is assigned points according to the origin of the BLAST-matches, as well as the origin predicted by HMMER. By default, the scoring system gives 5 points to the origin predicted by the HMMER-based extraction. The origin of the best BLAST-match to the sequence is also given 5 points, and the origins of the subsequent BLAST-matches is given scores decreasing by one for each match. When all BLAST-matches have been analysed, the score for each origin is summed up, and the sequence is assigned to the origin with the highest score. If the origin of the final classification does not agree with the predicted origin from the HMMER-based step, the sequence classification is marked as uncertain (by applying a “#” to the end of the definition line). The sequence is also marked as uncertain if the difference between the scores of the two most likely origins is smaller than the number of sequences of analysed BLAST-matches (by default 5).

The second classification step makes Metaxa very accurate even on fragmentary sequences, by ensuring that two independent methods agree on the predictions made. By applying stringent criteria in the extraction step, the software is still very robust with respect to false positives, and also reasonably fast, even on large metagenomic data sets. Its performance is

slower on large PCR-libraries, however, as more of the sequences will represent the SSU and hence need to be classified in the second step.

While Metaxa's default settings should be usable in most situations, you should consider if they are suitable for your purposes and for your data set. If the data set is small, this can be done by running the software multiple times on the data, with different settings, and analyse the outcome. On larger data sets, it might be more feasible to only run Metaxa on a sub set of the sequences for testing. The graphical output is very useful for determining whether Metaxa performs as desired on the data, as the positions of the found conserved domains can be easily investigated. If domains are missing, the criteria might be set to be too stringent. If they are not in sequential order (from V11 to V9r), that might be an indication that there is something wrong with the input sequences.

The HMMER program `hmmscan`, used by Metaxa, normally uses heuristic filters to increase the search speed. Metaxa runs `hmmscan` with the `--max` option in order to turn off all heuristic filters. This increases detection power at the cost of speed. However, the time requirement of the HMMER search is generally not an issue with Metaxa, while accuracy is, and thus the heuristic filters are not used.

## 5. Running Metaxa's analysis steps separately

Metaxa's analysis procedure is divided into two steps, the extraction and the classification. These two steps are normally run in sequence by running the `metaxa` command. However, they can also be run separately if the user wishes. To run the extraction step independently, use the `metaxa_x` command. This command takes a subset of the `metaxa` options (other options will be ignored). To see the available options for the `metaxa_x` command, type `"metaxa_x --help"` on the command line. To run the classification step on a set of SSU sequences, use the command `metaxa_c`. The options for `metaxa_c` can be seen by typing `"metaxa_c --help"` on the command line. Note that the output files obtained when running each step separately will be slightly different than obtained through running the entire Metaxa pipeline.

## 6. 'Undocumented' features

Metaxa has two undocumented options that can be activated, but they are considered *experimental* and should be used with caution. One allows you to use pre-calculated `hmmscan` results and feed into Metaxa, the other allows using a set of additional HMM-profiles for the SSU extraction.

### 'Undocumented' options:

- |                               |  |
|-------------------------------|--|
| <code>--hmmscan {file}</code> | If the <code>hmmscan</code> has already been performed, this option can be used as the base for the <code>hmmscan</code> output files, and the <code>hmmscan</code> step will be skipped. Overrides the <code>-o</code> option, while a DNA FASTA input file (containing the sequences used for the <code>hmmscan</code> ) must still be supplied! This feature is pretty experimental and was used during early evaluation of Metaxa. Use it only with caution. |
| <code>-t {o, other}</code>    | It is possible to supply an additional set of HMM-profiles in an <code>O.hmm</code> file within the HMMs directory. This custom set can be any type of profiles, but the profiles must be named according to the convention in the other HMM-files, beginning with V11 and ending with V9r.  |

## **7. License information**

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but **WITHOUT ANY WARRANTY**; without even the implied warranty of **MERCHANTABILITY** or **FITNESS FOR A PARTICULAR PURPOSE**. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program, in a file called 'license.txt'. If not, see: <http://www.gnu.org/licenses/>.

Copyright (C) 2011 Johan Bengtsson et al.