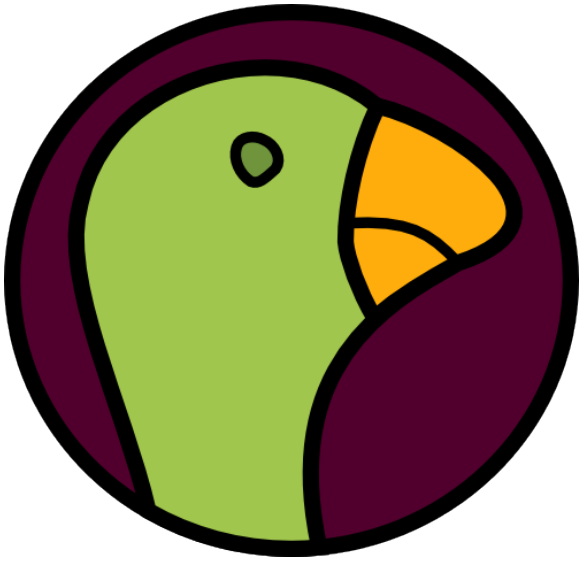

Plant & Food RESEARCH
RANGAHAU AHUMĀRA KAI

Moa

author: Mark Fiers

date: Sat Dec 26 11:02:34 NZDT 2009

git: master 3d1e84adf6f3b9e6427b4eae1a465d0addc88889

Contents

Chapter 1

Introduction

NOTE: both the software and the manual are under heavy development. Expect things to change.

Moa is a set of tools build around GNU make[4] that facilitates the use of GNU make in bioinformatics data analysis.

GNU Make is developed to aid in compilation of software. Software compilation usually involves the execution of many preprocessing, compilation and linker steps, with different parameters and interdependent of each other.

Gnu make is able to compile tens of thousands of files in large software projects through a detailed description of exactly what target files are to be created; from what source files; in what order; and using which libraries. If, during development, a few source files have changed, Gnu Make is able to repeat only the affected part of the build process.

The description, used by Gnu Make, that describes the build process is called a Makefile. The syntax of a Makefile is flexible enough to allow Gnu Make to be used for practically any programming language. Moreover, Gnu Make can be used to automate any series of commands (as long as they can be executed from the command line). It is therefore not only possible, but an excellent idea (not mine), to use Gnu Make in bioinformatics projects (see: X, Y, Z)

A bioinformatics analysis is often a set of interdependent, standard, steps (rather like compiling software). For example: (1) You take a piece of genomic DNA; (2–4) perform a set of gene predictions; (5) integrate the predictions and (6) run BLAST on the predicted genes.

There are, apart from using Makefiles, many different ways to automate this [refs..], each with its advantages. A surprising number of bioinformaticians, however, use either the command line or small, tailor-made, scripts to retain ultimate flexibility. Using scripts has as advantage that it is easy to repeat an

analysis by rerunning the script. Such a script could be written in any language (Bash, Perl, Python) but could also be a custom Makefile.

. . .

Moa wraps a set of common bioinformatics tools as Makefiles. Using Moa gives you a:

- A uniform interface; although Moa is based around Gnu Make, all commands are executed using the "moa" utility script. The "moa" script often just invokes Gnu Make but is able to handle a few extra cases where the use of Gnu Make is not possible.
- An easy way to track and repeat a set of analyses.
- Interaction; the Makefile templates are designed to interact with each other and make it easy to build pipelines with the Moa makefiles as building blocks.
- Parallel execution; Gnu make facillitates (limited) parallel execution of jobs. There is nothing however, that prevents integrations with a third party cluster solution such as Hadoop or SGE.

1.0.1 Example session

The best way to understand how to use Moa is a sample session.

We'll start by creating directories to hold the data and analysis structure:

```
mkdir introduction
cd introduction
```

We've created a directory called `introduction` to store the introductory tutorial. Within this directory we'll organize the components of our sample analysis. We want to initialize this directory so that it becomes a part of a moa pipeline. This is usefull later, if we want to run all analysis at once. To do this, run:

```
moa new -p introduction
```

The "moa new" command is used to create new moa jobs. In this case, since it is the first the -p (or—project) parameter tells Moa that this project is called "introduction". Moa uses a frontend script (called moa) to provide uniform interaction with the system. We'll now create a new directory to hold the first step of the pipeline:

```
mkdir 10.download
cd 10.download
moa new
```

Moa doesn't enforce any organization of an analysis pipeline, but expects the user to do so. An easy way to do this is by employing a logical directory structure. Hence, the directory describing the first step in our analysis: downloading data, is prefixed with a `10.`. Later steps will use higher numbers. Note that "moa new" is executed again, this time omitting the -p parameter. If the project parameter is omitted, moa tries to resolve this by reading the moa configuration in the parent directory.

We will now created a new folder to hold a genome sequence we are about to download and set up the Moa makefile to actually do the download.

```
mkdir 10.genome
cd 10.genome
moa new -t 'download a potato bac' ncbi
```

This time we have added a new parameter to the 'moa new' invocation: "ncbi". This tells Moa that in this directory the "ncbi" template should be used that allows easy downloading of information from NCBI. We also provide, as a good practice, a descriptive title using the -t (or—title) parameter. In general, once a moa makefile is instantiated you can call "moa help" to get some information on how to use this template:

```
moa help
```

(Note that if you want help on how to use the moa frontend script, you should use moa—help)

Before you can execute this job you have to tell what needs to be downloaded. This is easy if you know the Genbank accession number. In this case we'll download the nucleotide sequence (from the database nuccore) with the accession id AC237669.1

moa set ncbi_db=nuccore moa set ncbi_query=AC237669.1

Moa will give a response indicating that it has set the two parameters. You can also check the "moa.mk" file that stores job specific parmaters or run:

moa show

helpgives you an overview of all the parameters that you can set. In the case of ann
set: make set jid=lactobacillus.genome This job is set up and can be executed

Chapter 2

Installation

2.1 Prerequisites

Moa is developed on Ubuntu[8] and RHEL[5] Linux and is expected to operate without much problems on most modern Linux distributions. Moa is depends on the following list of software. The version numbers are an indication, not strict prerequisites. Other, even older, versions might work.

- Gnu Make 3.81
- Git 1.6. To download the Moa software. Alternatively it is possible to download a tarball.
- Python 2.6. Python version 2.5 and lower will not work, several supporting scripts use 2.6 specific functionality
- Bash. Many of the embedded scripts expect the Bash shell. Luckily, Bash is the default shell of almost all Linux distributions.
- Gnu Make Standard Library (GSML). A set of standard routines for Gnu Make. GSML is embedded in this distribution.

2.1.1 Couchdb

Moa can use Apache's Couchdb as a central storage of information on Moa jobs. allowing other Moa jobs to refer hereto. If you want to use this, the following prerequisites are added to the list:

- Apache Couchdb 0.9.0. Only when using couchdb functionality, see the chapter on Couchdb

- Couchdb-python. Only when using couchdb functionality, see the chapter on Couchdb

For more information, read the chapter on couchdb.

2.1.2 Bioinformatics tools

Each of the wrapped tools, obviously, requires that these tools are present. Usually, unless mentioned otherwise, Moa expects all tools to be installed in the system PATH. All requirements are described in the reference chapter.

2.1.3 Deciding where to install Moa

You will need to choose a location to install Moa to, this usually depends on who is going to use the software. Moa can be installed system wide for all users of this machine, for example in `/opt/moa`. However, if you will be the only person using Moa, install it in your home directory, for example under `~/moa`. The remainder of this chapter assumes an installation in your home directory.

2.2 Downloading Moa

Moa is hosted at github:

`http://github.com/mfiers/Moa`

Currently there are no stable releases so the best option is to download the latest version of the software, this can be done using Git or by downloading a source archive.

2.2.1 Using Git

Using git is a good choice as long as there are no releases. Git makes it very easy to stay up to date with the latest version and, even better, allows anybody to submit bugfixes to the Moa repository (more on that later). To download Moa using Git, enter the following commands (assuming you're installing Moa in your home directory):

```
cd ~
git clone git://github.com/mfiers/Moa.git moa
```

2.2.2 Downloading an archive

As an alternative, it is possible to download an (automatically generated) archive of the latest Moa version here, for example, using the following commands:

```
wget http://github.com/mfiers/Moa/tarball/master
```

The archive that is downloaded will have a rather long name that looks something like `mfiers-Moa-b13ddf78c6a1ae9a714c7d9979a1b1de0ed08462.tar.gz`. This archive needs to be unpacked in a temporary directory and then moved to its final location:

```
mkdir /tmp/moa_install
cd /tmp/moa_install
tar xvzf mfiers-Moa-b13ddf78c6a1ae9a714c7d9979a1b1de0ed08462.tar.gz
mv mfiers-Moa-b13ddf78c6a1ae9a714c7d9979a1b1de0ed08462.tar.gz ~/moa
```

After following either procdure; downloading the archive or using Git, the source code tree should be should be in its final location. The tree should contain the following directories:

```
./moa
    ./bin
    ./doc
    ./etc
    ./lib
    ./template
    ./test
    ./util
    ./www
    ./COPYING
    ./INSTALL
    ./README
    ./VERSION
```

2.3 Configuration

Configuration of Moa is simple: The Moa `/bin/` directory must be included in the PATH and an environment variable must be set pointing to the Moa directory. The easiest way to do this is by adding the following lines to your `.bashrc`:

```
export PATH=/opt/moa/bin:$PATH
export MOABASE=/opt/moa
```

and run:

```
source .bashrc
```

Also, if you are running Moa to be used by all users of your system system, please remember the file attributes correctly:

```
chmod a+rX -R $MOABASE
chmod a+rx $MOABASE/bin/*
```

Chapter 3

Using Moa

3.1 Creating a pipeline

3.1.1 Guiding principles

Most (bioinformatics?) projects start small, and grow over time. From that perspective it is advisable to give the organization of your project some thought on forehand.

When using Moa the separate analysis steps of a pipeline each reside in a directory. The output data of each analysis usually resides in the same directory or a subdirectory thereof. Moa has templates that assist in downloading and organizing data. This has as result that all project data in a Moa project will be organized in a directory tree on your filesystem. Such a tree must represent both the data in logical way as well as the analysis pipeline organization.

Although there are likely multiple ways of achieving a healthy organization of a Moa project, this manual proposes the following organization:

- On the highest levels organize your project according to fundamental divisions in the project or data source. For example, if you work with data from multiple organisms, that might be a good top level division.
- On lower levels start organizing your annotation pipeline. Since most

3.1.2 Setting up new jobs - `moa new`

Creating a new job is done with the `moa new` command.

3.2 Running a pipeline

3.2.1 Running one job

3.2.2 Running a series of jobs

Chapter 4

Using GBrowse

The Generic Genome Browser (Gbrowse)[7] is a popular tool for ...

to be written

Chapter 5

Extending Moa

This chapter describes how to create new templates for use with Moa. Creating a template is not very difficult, once you have a basic understanding of how Makefiles work. Pprobably the hardest part is ensuring that templates are able to interact with other templates.

A template is, as stated, not much more than Makefile that adheres to certain standards. To understand how Makefiles work, please read the Gnu Make Manual. Note that creating Makefiles can be somewhat complex at first, given that the logic differs from scripting languages. The easiest way to do this is to work from an existing Makefile.

Each template exists of the following parts:

- Definition
- Include moaBase
- Implementation

The order in which everything is defined in a template is very important! It is advisable to not define variables depending on other variables in the definition phase.

In the remainder of this chapter we will describe a simple template that creates the reverse complement of a FASTA file using the EMBOSS[6] revseq utility

5.1 Definition

The definition is a list of variables defining what your template does and giving Moa information on how to use this template.

5.1.1 Describing the new template

The following variables define what your template does. These variables are used in generating the help files, the manual and the website.

Identifier	Description
moa_title	The title for this template
moa_description	A short description of this template
moa_ids	A unique, short, identifier for this template

Example:

```
moa_title = Reverse Complement
moa_description = This Moa template takes a set of      \
    input FASTA sequences and determines the reverse    \
    complement using the EMBOSS revseq utility.
moa_ids += revcom
```

Note that lines are allowed to break over multiple lines, given that each line that continues to the enxt line ends with a backslash. No spaces are allowd after the backslash and the new line must be indented (with at least one space).

5.1.2 Moa organizational units - moa_ids

In the previous chapter, both title and description are fairly self evident. The `moa_ids` variable is, however, more complicated. Each template must have, at least one, unique, preferaby short, identifier linked to it. This `moa_id` helps in defining variable space for each template. The moa_id returns when defining template specific variables and targets. All template specific variables have the moa_id as a part of their name, so do the major targets of a template.

Use of uniqued ids allow Moa to stack several templates into a larger, more complicated, templates. This might be usefull describing a set of resembling tasks that have a lot of overlapping code. Another powerful use is to create complex jobs that execute a mini-pipeline in one run. For example, gathering a filter a specific set of sequences (using the gather template) and creating a BLAST database from that.

Using ids allows functional sepeartion of tasks withing a template, or within a stacked template. It is advisable to start creating templates with only one task. For each task, a set of specific variables need to be defined.

Given that a template can define multiple tasks, a moa_id are added to the moa_ids array using the following syntax:

```
moa_ids += revcomp
```

5.1.3 taks specfic variables

Identifier	Description
moa_title	The title for this template
moa_description	A short description of this template
moa_ids	A unique, short, identifier for this template

5.2 Include moaBase

To includie moaBase add the following line to your Makefile:

```
include $(shell echo $$MOABASE)/template/moaBase.mk
```

5.3 Implementation

5.3.1 define dependant variables

5.3.2 define targets

Each task, identified by a unique moa_id, needs to define a set of four targets. For example, if your template defines: `moa_id += revomp` then the following four targets are expected to be defined and are automatically executed:

- MOA_ID - revcomp
- MOA_ID_prepare - revcomp_prepare
- MOA_ID_post - revcomp_post
- MOA_ID_clean - revcomp_clean

Each of these targets must be defined in a new template, although they could can be empty. In the following paragraphs, each of these targets are discussed, in the order that they are executed.

Prepare execution: revcomp_prep

The MOA_ID_prep target contains commands that are executed prior to the main run. In the case of reverse complementing sequences this target can be used to create a directory to store the output sequences. Using a separate subdirectory to

Round up execution: revcomp_post

Clean up: revcomp_clean

- **revcomp**: the main target, executes the main task of this template. In this case it takes a set of input sequences and write the reverse complement back to disk.
- **revcomp_prepare**:
- **revcomp_post**: Optional commands to be exeucted after everything is finished. In the case of reverse complementing a set of sequences there is not much to do. The BLAST template, however, uses this target to create an overall BLAST report
- **revcomp_clean**: Cleans up all reverse complemented sequences

Appendix A

Template reference

This chapter contains, as a reference, all help documentation of all templates currently in the Moa repository. It is possible to get the (latest) version of the help for each template by running:

```
make help
```

in a directory with a Moa analysis. Moreover, if an Makefile links to multiple templates, on the fly generated help will detail all targets that can be used and all parameters that can be defined.

A.1 act_crunch

Discover the bidirectional best blast hit between two sets of sequences

A.1.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (crunch)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

crunch generate a list of bidirectional best blast hits.

A.1.2 Parameters

Required parameters

crunch_input_fila_a First multifasta input file (file)

crunch_input_fila_b First multifasta input file (file)

title A job name - Describe what you are doing (string)

Optional parameters

crunch_protein Are we looking at proteins? (T|*F*)

crunch_eval e value cutoff (float, default:1e–10)

crunch_nothreads threads to run crunch with (note the overlap with the Make -j parameter) (integer, default:4)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.2 bidibebla

Discover the bidirectional best blast hit between two sets of sequences

A.2.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (bdbb)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

bdbb generate a list of bidirectional best blast hits.

A.2.2 Parameters

Required parameters

bdbb_input_fila_a First multifasta input file (file)

bdbb_input_fila_b First multifasta input file (file)

title A job name - Describe what you are doing (string)

Optional parameters

bdbb_protein Are we looking at proteins? (T|*F*)

bdbb_eval e value cutoff (float, default:1e–10)

bdbb_nothreads threads to run bdbb with (note the overlap with the Make -j parameter) (integer, default:4)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.3 blast

Wraps BLAST[1], the most popular similarity search tool in bioinformatics

A.3.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (blast)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

blast Running BLAST takes an input directory (*blast_input_dir*), determines what sequence files are present (with the parameter *blast_input_extension*) and executes BLAST on each of these. Moa BLAST is configured to create XML output (as opposed ot the standard text based output) in the *./out* directory. The output XML is subsequently converted to GFF3[3] by

the custom *blast2gff* script (build around biopython[2]). Additionally, a simple text report is created.

blast_report Generate a text BLAST report.

A.3.2 Parameters

Required parameters

blast_input_dir Directory with the BLAST input files (directory)

blast_db Location of the blast database. You can either define the blast db parameter as used by blast, or any of the blast database files, in which case the extension will be removed before use (file)

title A job name - Describe what you are doing (string)

Optional parameters

blast_input_extension file extension for the files in blast_input_dir (string, default:fasta)

blast_input_glob glob to select a subset of files from blast_input_dir (string, default:*)

blast_gff_source source field to use in the gff (string, default:BLAST)

blast_program blast program to use (default: blastn) (blastx|*blastn*|blastp|t*blastn*|tblastx)

blast_eval e value cutoff (float, default:1e–10)

blast_nohits number of hits to report (integer, default:50)

blast_nothreads threads to run blast with (note the overlap with the Make -j parameter) (integer, default:2)

blast_gff_blasthit (T,**F**) - export an extra blasthit feature to the created gff, grouping all hsp (match) features. (T|*F*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.4 blastSingle

Wraps BLAST[1], the most popular similarity search tool in bioinformatics

A.4.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (blast)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

blast Running BLAST takes an input directory (*blast_input_dir*), determines what sequence files are present (with the parameter *blast_input_extension*) and executes BLAST on each of these. Moa BLAST is configured to create XML output (as opposed ot the standard text based output) in the *./out* directory. The output XML is subsequently converted to GFF3[3] by the custom *blast2gff* script (build around biopython[2]). Additionally, a simple text report is created.

blast_report Generate a text BLAST report.

A.4.2 Parameters

Required parameters

blast_input_file Input fasta file to BLAST (file)

blast_input_dir Directory with the BLAST input files (directory)

blast_db Location of the blast database. You can either define the blast db parameter as used by blast, or any of the blast database files, in which case the extension will be removed before use (file)

title A job name - Describe what you are doing (string)

Optional parameters

blast_input_extension file extension for the files in blast_input_dir (string, default:fasta)

blast_input_glob glob to select a subset of files from blast_input_dir (string, default:*)

blast_gff_source source field to use in the gff (string, default:BLAST)

blast_program blast program to use (default: blastn) (blastx|*blastn*|blastp|t*blastn*|tblastx)

blast_eval e value cutoff (float, default:1e–10)

blast_nohits number of hits to report (integer, default:50)

blast_nothreads threads to run blast with (note the overlap with the Make -j parameter) (integer, default:2)

blast_gff_blasthit (T,**F**) - export an extra blasthit feature to the created gff, grouping all hsp (match) features. (T|*F*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.5 blastdb

A.5.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (blastdb)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

blastdb Takes either a set of fasta files or a single multi-fasta input file and creates a BLAST database.

A.5.2 Parameters

Required parameters

bdb_name Database name to create. (string)

title A job name - Describe what you are doing (string)

Optional parameters

bdb_input_dir Dir with the input fasta files, defaults to ./fasta (directory)

bdb_input_extension extension of the input sequence files, defaults to fasta (string, default:fasta)

bdb_fasta_file The file with all FASTA sequences for the blastdb concatenated. This can be used as an alternative to defining bdb_input_dir and bdb_input_dir_extension. Morover. If all your sequences are already in a single file, then using this parameter prevents duplication of that file. (file)

bdb_protein Protein database? (T)rue) or not (F)alse (default: F) (T|*F*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.6 blat

A.6.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (blat)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

blat :

A.6.2 Parameters

Required parameters

blat_db type of the database (dna, prot or dnax) (['dna',|'prot',|'dnax']**)

blat_input_file input query file. If this variable is not defined, the combination of blat_input_dir and blat_input_extension is used to find a list of input files (file)

blat_input_dir source field in the generated gff (directory)

blat_gff_source Source field for the generated GFF files (string)

title A job name - Describe what you are doing (string)

Optional parameters

blat_input_extension extension of the input files (string, default:fasta)

blat_eval evalue cutoff to select the reported hits on (defaults to 1e–15) (float, default:1e–10)

blat_db_id_list a sorted list of db ids and descriptions, enhances the report generated (file)

blat_db_type type of the database (dna, prot or dnax) (['*dna*',|'prot',|'*dnax*'])

blat_query_type type of the query (dna, rna, prot, dnax or rnax) (['*dna*',|'rna',|'prot',|'*dnax*',|'rnax'])

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.7 bowtie

A.7.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (bowtie)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

bowtie :

A.7.2 Parameters

Required parameters

bowtie_db The bowtie database to use. It is allowed to define one of the bowtie database files (.[0–9].ebwt). (file)

bowtie_input_dir Input files for bowtie (directory)

title A job name - Describe what you are doing (string)

Optional parameters

bowtie_input_extension file extension for the files in bowtie_input_dir (string, default:fastq)

bowtie_input_glob glob to select a subset of files from bowtie_input_dir (string, default:*)

bowtie_input_format Format of the input files (*fastq*|fasta)

bowtie_extra_params extra parameters to feed bowtie (string)

bowtie_paired_ends perform a paired end analysis. If so, the input files are expected to be of the form '*_1.fastq' and ' *_2.fastq' (T|*F*)

bowtie_forward_suffix Last part of the sequence name identifying a file with forward reads (string, default:_1)

bowtie_reverse_suffix Last part of the sequence name identifying a file with reverse reads (string, default:_2)

bowtie_output_format Format of the output file (bowtie|*bam*|sam)

bowtie_insertsize Expected insertsize (float, default:5000)

bowtie_insertsize_sed SED expression to filter the expected insertsize from the input file name (string)

bowtie_insertsize_min multiplier determining the minimal acceptable value for two paired reads to be apart. If the bowtie_insertsize is 10000 and this parameter is set at 0.8, than reads that are closer together than 8000 nt are rejecte (float, default:0.1)

bowtie_insertsize_max Max insertsize for a paired alignment (float, default:10)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.8 bowtiedb

A.8.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (bowtiedb)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

bowtiedb :

A.8.2 Parameters

Required parameters

bowtiedb_input_dir Sequence files used to build a bowtie database (directory)

bowtiedb_name Name of the bowtie index to create (string)

title A job name - Describe what you are doing (string)

Optional parameters

bowtiedb_input_extension file extension for the files in bowtiedb_input_dir (string, default:fasta)

bowtiedb_input_glob glob to select a subset of files from bowtiedb_input_dir (string, default:*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.9 cleanFasta

A.9.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (clean_fasta)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

clean_fasta Cleanup of a FASTA file (in place!)

A.9.2 Parameters

Required parameters

cf_input_dir Directory with the sequences to run cleanfasta on (directory)

title A job name - Describe what you are doing (string)

Optional parameters

cf_input_extension input file extension (string, default:fasta)

sed_command The sed command cleaning the code, defaults to '/>/!s/[ACGTNacgtn]/N/g' (string, default:/>/!s/[ACGTNacgtn]/N/g)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.10 clustalgroup

A.10.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (clustalgroup)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

clustalgroup run clustalw

A.10.2 Parameters

Required parameters

cwg_input_dir This set of sequences to run clustalw on (directory)

title A job name - Describe what you are doing (string)

Optional parameters

cwg_input_extension Input file extension (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.11 clustalpair

A.11.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (clustalpair)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

clustalpair run clustalw

A.11.2 Parameters

Required parameters

input_dir_a This set is compared to the sequences in input_dir_b. only a forward comparison is made (a against b, not the other way round) (directory)

input_dir_b The set to compare against (directory)

title A job name - Describe what you are doing (string)

Optional parameters

input_extension Extension of the input files (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.12 clustalw

A.12.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (clustalw)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

clustalw run clustalw

A.12.2 Parameters

Required parameters

input_dir_a This set is compared to the sequences in input_dir_b. (directory)

input_dir_b The set to compare against. Only a forward comparison is made (a against b, not the other way round) (directory)

title A job name - Describe what you are doing (string)

Optional parameters

input_extension Extension of the input files (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.13 concatenate

A.13.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (concatenate)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

concatenate Concatenate a set of FASTA files

A.13.2 Parameters

Required parameters

input_dir Directory with the input data (directory)

name name of the file, the outputfile will become ./name.fasta (string)

title A job name - Describe what you are doing (string)

Optional parameters

input_extension Extension of the input files (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.14 create.gbrowse.db

A library that aids in uploading FASTA and GFF to a Generic Genome Browser database. This template is only to be used embedded in another template. This library expects that the following variables are preset; gup_fasta_dir, gup_gff_dir gup_upload_fasta, gup_upload_gff

A.14.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (upload2gbrowse)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

upload2gbrowse :

initGbrowse Clean & initalize a gbrowse database. **Warning: all data will be lost!**

gupgo Actually do the upload. upload2gbrowse NEVER does this automatically!

A.14.2 Parameters

Required parameters

gup_user gbrowse db user. If not defined, this defaults to 'moa'.

gup_db gbrowse database. If not defined, this defaults to 'moa'.

title (string) A job name - Describe what you are doing

Optional parameters

gup_gff_extension extension of the GFF files to upload (.gff)

gup_fasta_extension extension of the FASTA files to upload (.fasta)

gup_upload_fasta upload fasta to gbrowse (T/F)

gup_upload_gff upload gff to gbrowse (T/F)

gup_force_upload upload to gbrowse, ignore gup_lock and upload all, not only files newer that upload_gff or upload_fasta

marks_extensions Add some extensions to the Gbrowse database to be initalized, for use by Mark.

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.15 crunch

Create a crunch file for use with the Artemis ACT comparison tool.

A.15.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (crunch)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

crunch create crunch files

A.15.2 Parameters

Required parameters

crunch_input_dir Directory with input fasta files (directory)

title A job name - Describe what you are doing (string)

Optional parameters

crunch_input_extension file extension for the files in crunch_input_dir (string, default:fasta)

crunch_input_glob glob to select a subset of files from crunch_input_dir (string, default:*)

crunch_eval e value cutoff (float, default:1e–10)

crunch_nothreads threads to run crunch with (note the overlap with the Make -j parameter) (integer, default:4)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.16 dottup

A.16.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (dottup)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

dottup Run dottup

A.16.2 Parameters

Required parameters

dottup_input_dir_a This set is compared to the sequences in input_dir_b. (directory)

dottup_input_dir_b The set to compare against (directory)

title A job name - Describe what you are doing (string)

Optional parameters

dottup_input_extension Extension of the dottup input files (string, default:fasta)

dottup_wordsize Wordsize used to discover similarities between sequences (integer, default:8)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.17 dottupSelf

A.17.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (dotself)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

dotself run clustalw

A.17.2 Parameters

Required parameters

dotself_input_dir Set of sequences to use (directory)

title A job name - Describe what you are doing (string)

Optional parameters

dotself_input_extension Extension of input files (string, default:fasta)

dotself_wordsize Wordsize used for recognizing similarity (integer, default:6)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.18 empty

A.18.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (empty)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

empty :

A.18.2 Parameters

Required parameters

title (string) A job name - Describe what you are doing

Optional parameters

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.19 fasta2gff

Derive GFF from a FASTA file, usually to accompany the Sequence for upload to a generic genome browser database.

A.19.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (fasta2gff)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

fasta2gff :

A.19.2 Parameters

Required parameters

f2g_gffsource Source to be used in the gff (string)

f2g_input_dir Directory with the input fasta files (directory)

title A job name - Describe what you are doing (string)

Optional parameters

f2g_output_dir Directory with the output gff (directory, default:./gff)

f2g_input_extension glob pattern of the fasta files (default: *.fasta) (string, default:fasta)

f2g_options options to be passed to the fasta2gff script (string)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.20 gap4export

A.20.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (gap4export)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

gap4export Export data from an assembly using gap4

A.20.2 Parameters

Required parameters

ge_input_dir Directory with the input data (directory)

ge_input_pattern file name pattern (string)

title A job name - Describe what you are doing (string)

Optional parameters

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.21 gather

A.21.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (gather)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

gather gather files

A.21.2 Parameters

Required parameters

g_input_dir list of directories with the input files (directory)

title A job name - Describe what you are doing (string)

Optional parameters

g_input_pattern glob pattern to download (string, default:*)

g_name_sed SED expression to be executed on each file name - allows you to change file names (string, default:s/a/a/)

g_output_dir Output subdirectory, defaults to '.' (directory, default:.)

g_parallel allow parallel execution (T) or not (**F**). If for example concatenating to one single file, you should not have multiple threads. (T|*F*)

g_process Command to process the files. If undefined, hardlink the files. (string, default:ln -f $<)

g_limit limit the number of files gathered (with the most recent files first, defaults to 1mln) (integer, default:1000000)

g_powerclean Do brute force cleaning (T/F). Remove all files, except moa.mk & Makefile when calling make clean. Defaults to F. (T|F)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.22 genemarks

predict genes using geneMarkS

A.22.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (genemarks)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

genemarks :

A.22.2 Parameters

Required parameters

genemarks_input_dir directory containing the input sequences (directory)

genemarks_matrix the matrix to use (file)

title A job name - Describe what you are doing (string)

Optional parameters

genemarks_gff_source source field to use in the gff. Defaults to geneMarkS (string, default:genemarks)

genemarks_input_extension input file extension. Defaults to 'fasta' (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.23 getFromNcbi

Download a set of sequences from NCBI based on a query string (ncbi_query) and database (ncbi_db). This tempate will run only once (!), after a succesful run it creates a 'lock' file that you need to remove to rerun

A.23.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (getFromNcbi)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

getFromNcbi Downloads from NCBI

A.23.2 Parameters

Required parameters

ncbi_query NCBI query (for example txid9397[Organism%3Aexp]) (string)

title A job name - Describe what you are doing (string)

Optional parameters

ncbi_db NCBI database (string, default:nuccore)

ncbi_sequence_name Sequence name to download. When this parameter is set, the template assumes that only one sequence is to be downloaded, the rest will be discarded. (string)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.24 getorf

Predicts open reading frames using the EMBOSS[?] getorf tool.

A.24.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (getorf)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

getorf :

A.24.2 Parameters

Required parameters

getorf_input_dir Input files for getorf (directory)

title A job name - Describe what you are doing (string)

Optional parameters

getorf_input_extension file extension for the files in getorf_input_dir (string, default:fasta)

getorf_input_glob glob to select a subset of files from getorf_input_dir (string, default:*)

getorf_gff_source source field to use in the gff. (string, default:getorf)

getorf_minsize minimal nucleotide size of the predicted ORF. (integer, default:30)

getorf_maxsize maximal nucleotide size of the predicted ORF. (integer, default:1000000)

getorf_circular Is the sequence linear? (Y|*N*)

getorf_table Genetic code to use: 0 Standard; 1 Standard with alternative initiation codons; 2 Vertebrate Mitochondrial; 3 Yeast Mitochondrial; 4 Mold, Protozoan, Coelenterate Mitochondrial and Mycoplasma/Spiroplasma; 5 Invertebrate Mitochondrial; 6 Ciliate Macronuclear and Dasycladacean; 9 Echinoderm Mitochondrial; 10 Euplotid Nuclear; 11 Bacterial; 12 Alternative Yeast Nuclear; 13 Ascidian Mitochondrial; 14 Flatworm Mitochondrial; 15 Blepharisma Macronuclear; 16 Chlorophycean Mitochondrial; 21 Trematode Mitochondrial; 22 Scenedesmus obliquus; 23 Thraustochytrium Mitochondrial. (0|1|2|3|4|5|6|7|8|9|10|*11*|12|13|14|15|16|21|22|23)

getorf_find What to output? 0: Translation between stop codons, 1: Translation between start & stop codon, 2: Nucleotide sequence between stop codons; 3: Nucleotide sequence between start and stop codons. Default: 3 (0|1|2|*3*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.25 glimmer3

Predicts (prokaryotic) using glimmer3.

A.25.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (glimmer3)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

glimmer3 Glimmer3 is a open reading frame discovery program from the EMBOSS[?] package. It takes a set of input sequences and predicts all open reading frames. Additionally, this template converts the default output (predicted protein sequences) to GFF3.

A.25.2 Parameters

Required parameters

glimmer3_input_dir Input directory with the sequences to run glimmer3 on (directory)

title A job name - Describe what you are doing (string)

Optional parameters

glimmer3_gff_source source field to use in the gff. Defaults to glimmer3 (string, default:glimmer3)

glimmer3_input_extension input file extension. Defaults to 'fasta' (string, default:fasta)

glimmer3_max_overlap Maximum overlap, see the glimmer documentation for the -o or—max_olap parameter (integer, default:50)

glimmer3_gene_len Minimum gene length (glimmer3 -g/—gene_len) (integer, default:110)

glimmer3_treshold treshold for calling a gene a gene (glimmer3 -t) (integer, default:30)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.26 gmap

A.26.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (gmap)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

gmap :

A.26.2 Parameters

Required parameters

gmap_db Gmap db (file)

gmap_input_file input file with the sequences to map (file)

title A job name - Describe what you are doing (string)

Optional parameters

gmap_extra_parameters extra parameters to feed to gmap (string)

gmap_invert_gff Invert the GFF (T/*F*) (T|*F*)

gmap_gff_source Source field to use in the output GFF (string, default:gmap)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.27 gmapdb

A.27.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (gmapdb)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

gmapdb :

A.27.2 Parameters

Required parameters

gmapdb_input_dir The reference sequence to build a gmap database with. (directory)

gmapdb_name Name of the gmap index to create (string)

title A job name - Describe what you are doing (string)

Optional parameters

gmapdb_input_extension Extension of the input files, defaults to 'fasta' (string, default:fasta)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.28 h_blast

Runs BLAST on a hadoop cluster

A.28.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (h_blast)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

h_blast Similar to a normal blast, but now running on an hadoop cluster

A.28.2 Parameters

Required parameters

hadoop_base location of the hadoop installation (directory)

h_blast_input_dir location of the hadoop installation (directory)

h_blast_db Location of the blast database (file)

title A job name - Describe what you are doing (string)

Optional parameters

hdfs_base htfs://SERVER:PORT for the hdfs filesystem, defaults to hdfs://localhost:9000 (string, default:hdfs://localhost:9000)

h_blast_input_extension input file extension (string, default:fasta)

h_blast_program blast program to use (default: blastn) (*blastn*|blastp|blastx|t*blastn*|tblastx)

h_blast_eval e value cutoff (float, default:1e–10)

h_blast_nohits number of hits to report (integer, default:50)

h_blast_nothreads threads to run blast with (note the overlap with the Make -j parameter) (integer, default:1)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.29 lftp

Use LFTP to download files. This template has two modi, one is set 'lftp_mode' to 'mirror' data, in which case both 'lftp_url' and 'lftp_pattern' (default *) are used. The other modus is 'lftp_mode=get', when one file defined by 'lftp_url' is downloaded. In the mirror mode it is possible to download only those files that are newer as the files already downloaded by using the 'lftp_timestamp' parameter

A.29.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (lftp)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

lftp execute the download

A.29.2 Parameters

Required parameters

lftp_url The base url to download from (string)

title A job name - Describe what you are doing (string)

Optional parameters

lftp_timestamp Depend on lftp to decide if a file needs updating, else a touch-file is created that you need to delete or touch before updating (T/*F*) (T|*F*)

lftp_powerclean Do brute force cleaning (T/F). Remove all files, except moa.mk & Makefile when calling make clean. Defaults to F. (T|*F*)

lftp_noclean set of files not to be deleted by the powerclean (string, default:moa.mk Makefile)

lftp_pattern glob pattern to download (string, default:'*')

lftp_lock Lock this job after running. This means that you will have to manually unlock the job before lftp actually reruns. This is a good choice if your downloading large datasets or have a slow connection (*T*|F)

lftp_user username for the remote site (string)

lftp_pass password for the remote site, note that this can be defined on the commandline using: 'make lftp_pass=PASSWORD' (password)

lftp_output_dir subdir to create & write all output to. If not defined, data will be downloaded to directory containing the Makefile (directory, default:.)

lftp_dos2unix Run dos2unix to prevent problems with possible dos text files (T|*F*)

lftp_mode Mode of operation - 'mirror' or 'get'. Mirror enables timestamping. Get just gets a single file. If using get, consider setting depend_lftp_timestamp to F. When using 'get', the full url should be in lftp_url. lftp_pattern is ignored. Defaults to mirror. (mirror|*get*)

lftp_get_name target name of the file to download (string)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.30 maq_fasta2bfa

A.30.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (f2b)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

f2b :

A.30.2 Parameters

Required parameters

f2b_input_dir (directory) input FASTA files

title (string) A job name - Describe what you are doing

Optional parameters

f2b_input_extension file extension for the files in f2b_input_dir (string, default:fasta)

f2b_input_glob glob to select a subset of files from f2b_input_dir (string, default:*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.31 maq_fastq2bfq

A.31.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (fq2bq)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

fq2bq :

A.31.2 Parameters

Required parameters

fq2bq_input_dir (directory) input FASTA files

title (string) A job name - Describe what you are doing

Optional parameters

fq2bq_input_extension file extension for the files in fq2bq_input_dir (string, default:fastq)

fq2bq_input_glob glob to select a subset of files from fq2bq_input_dir (string, default:*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.32 maq_match_pair

A.32.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (maq-pair)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

maqpair :

A.32.2 Parameters

Required parameters

maqpair_read_dir (string) directory containing the forward reads

maqpair_forward_suffix (string, default:_f.bfq) Suffix of each forward filename - recognize forward files this way. Note this is not a regular extension, no '.' is assumed between the filename & suffix

maqpair_reference (string) Reference bfa file to map the reads to

title (string) A job name - Describe what you are doing

Optional parameters

maqpair_reverse_suffix suffix of reverse files (string, default:_r.bfq)

maqpair_RF_maxdist max outer distance for an RF readpair (corresponds to the -A parameter). This applies to long insert illumina pairs (integer, default:15000)

maqpair_maxdist max outer distance for a (non RF) readpair. This applies to illumina matepairs - i.e. short inserts (integer, default:250)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.33 moatest

Not to be used - is used by unitmoatests

A.33.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (moatest)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

moatest Do nothing - no need to call this.

A.33.2 Parameters

Required parameters

txt test variable (string)

title A job name - Describe what you are doing (string)

Optional parameters

test_opt test variable (string, default:konijntje)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.34 mummer

A.34.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (mummer)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

mummer :

A.34.2 Parameters

Required parameters

mum_input_a_dir (directory) Set 1 input fasta files

mum_input_b_dir (directory) Set 1 input fasta files

title (string) A job name - Describe what you are doing

Optional parameters

mum_input_a_extension file extension for the files in mum_input_a_dir (string, default:fasta)

mum_input_a_glob glob to select a subset of files from mum_input_a_dir (string, default:*)

mum_input_b_extension file extension for the files in mum_input_b_dir (string, default:fasta)

mum_input_b_glob glob to select a subset of files from mum_input_b_dir (string, default:*)

mum_breaklen Set the distance an alignment extension will attempt to extend poor scoring regions before giving up (default 200) (integer, default:200)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.35 ncbi

Download a set of sequences from NCBI based on a query string (ncbi_query) and database (ncbi_db). This tempate will run only once (!), after a succesful run it creates a 'lock' file that you need to remove to rerun

A.35.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (getFromNcbi)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

getFromNcbi Downloads from NCBI

A.35.2 Parameters

Required parameters

ncbi_query NCBI query (for example txid9397[Organism%3Aexp]) (string)

title A job name - Describe what you are doing (string)

Optional parameters

ncbi_db NCBI database (string, default:nuccore)

ncbi_sequence_name Sequence name to download. When this parameter is set, the template assumes that only one sequence is to be downloaded, the rest will be discarded. (string)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.36 newbler

A.36.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (newbler)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

newbler :

A.36.2 Parameters

Required parameters

newbler_input_dir input SFF files (directory)

title A job name - Describe what you are doing (string)

Optional parameters

newbler_input_extension file extension for the files in newbler_input_dir (string, default:sff)

newbler_input_glob glob to select a subset of files from newbler_input_dir (string, default:*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.37 nstretch

A.37.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (nstretch)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

nstretch :

A.37.2 Parameters

Required parameters

nstretch_input_dir input dir with the fasta files (directory)

title A job name - Describe what you are doing (string)

Optional parameters

nstretch_input_extension extension of the input files (string, default:fasta)

nstretch_len minimal number of Ns before its reported (default 10) (integer, default:10)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.38 pregap

A.38.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (pregap)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

pregap Run pregap

A.38.2 Parameters

Required parameters

input_dir Directory with the input data (string)

input_pattern file name pattern (string)

cloning_vector File containing the cloning vector (file)

sequencing_vector File containing the sequencing vector (file)

ecoli_screenseq File containing ecoli screen sequences (file)

repeat_masker_lib File with a repeatmasker library (file)

vector_primerfile File with the vector primers (file)

title A job name - Describe what you are doing (string)

Optional parameters

quality_value_clip quality cutoff (integer, default:10)

pregap_template the template pregap config file to use. if not defined, Moa tries ./files/pregap.config. (file, default:./files/pregap.config.)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.39 project

Create a new project. All subdirectories of this directory are automatically a part of this project

A.39.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (project)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

project This template does not do anything - it is a project placeholder.

A.39.2 Parameters

Required parameters

title A job name - Describe what you are doing (string)

Optional parameters

project_description A short description of what this project is supposed to achieve, how to use it, and what parameters are most important to set (string)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.40 revseq

This Moa template takes a set of input FASTA sequences and determines the reverse complement using the EMBOSS revseq utility.

A.40.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (getorf)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

getorf :

A.40.2 Parameters

Required parameters

getorf_input_dir Input files for getorf (directory)

title A job name - Describe what you are doing (string)

Optional parameters

getorf_input_extension file extension for the files in getorf_input_dir (string, default:fasta)

getorf_input_glob glob to select a subset of files from getorf_input_dir (string, default:*)

getorf_gff_source source field to use in the gff. (string, default:getorf)

getorf_minsize minimal nucleotide size of the predicted ORF. (integer, default:30)

getorf_maxsize maximal nucleotide size of the predicted ORF. (integer, default:1000000)

getorf_circular Is the sequence linear? (Y|*N*)

getorf_table Genetic code to use: 0 Standard; 1 Standard with alternative initiation codons; 2 Vertebrate Mitochondrial; 3 Yeast Mitochondrial; 4 Mold, Protozoan, Coelenterate Mitochondrial and Mycoplasma/Spiroplasma; 5 Invertebrate Mitochondrial; 6 Ciliate Macronuclear and Dasycladacean; 9 Echinoderm Mitochondrial; 10 Euplotid Nuclear; 11 Bacterial; 12 Alternative Yeast Nuclear; 13 Ascidian Mitochondrial; 14 Flatworm Mitochondrial; 15 Blepharisma Macronuclear; 16 Chlorophycean Mitochondrial; 21 Trematode Mitochondrial; 22 Scenedesmus obliquus; 23 Thraustochytrium Mitochondrial. (0|1|2|3|4|5|6|7|8|9|10|*11*|12|13|14|15|16|21|22|23)

getorf_find What to output? 0: Translation between stop codons, 1: Translation between start & stop codon, 2: Nucleotide sequence between stop codons; 3: Nucleotide sequence between start and stop codons. Default: 3 (0|1|2|*3*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.41 sam2bam

A.41.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (sam2bam)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

sam2bam :

A.41.2 Parameters

Required parameters

sam2bam_input_dir (directory) input SAM files

title (string) A job name - Describe what you are doing

Optional parameters

sam2bam_input_extension file extension for the files in sam2bam_input_dir (string, default:sam)

sam2bam_input_glob glob to select a subset of files from sam2bam_input_dir (string, default:*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.42 traverse

Do nothing, except be a part in executing full directory structures

A.42.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (traverse)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

traverse Do nothing - no need to call this.

A.42.2 Parameters

Required parameters

title (string) A job name - Describe what you are doing

Optional parameters

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.43 upload2gbrowse

A library that aids in uploading FASTA and GFF to a Generic Genome Browser database. This template is only to be used embedded in another template. This library expects that the following variables are preset; gup_fasta_dir, gup_gff_dir gup_upload_fasta, gup_upload_gff

A.43.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (upload2gbrowse)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

upload2gbrowse :

initGbrowse Clean & initalize a gbrowse database. **Warning: all data will be lost!**

gupgo Actually do the upload. upload2gbrowse NEVER does this automatically!

A.43.2 Parameters

Required parameters

gup_upload_fasta upload fasta to gbrowse (T/F) (T|*F*)

gup_upload_gff upload gff to gbrowse (T/F) (T|*F*)

gup_user gbrowse db user. If not defined, this defaults to 'moa'. (string)

gup_db gbrowse database. If not defined, this defaults to 'moa'. (string)

title A job name - Describe what you are doing (string)

Optional parameters

gup_fasta_dir input directory with fasta files to upload to gbrowse (directory)

gup_gff_dir input directory with gff files to upload to gbrowse (directory)

gup_gff_extension extension of the GFF files to upload (.gff) (string, default:gff)

gup_fasta_extension extension of the FASTA files to upload (.fasta) (string, default:fasta)

gup_upload_fasta upload fasta to gbrowse (T/F) (T|*F*)

gup_upload_gff upload gff to gbrowse (T/F) (T|*F*)

gup_force_upload upload to gbrowse, ignore gup_lock and upload all, not only files newer that upload_gff or upload_fasta (T|*F*)

marks_extensions Add some extensions to the Gbrowse database to be initalized, for use by Mark. (T|*F*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.44 varscan

A.44.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (varscan)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

varscan :

A.44.2 Parameters

Required parameters

varscan_input_file Varscan input alignments file (file)

title A job name - Describe what you are doing (string)

Optional parameters

varscan_extra_params location of varscan.pl, defaults to '/usr/lib/perl5/site_perl/5.8.8/varscan.pl' (string)

varscan_output_name Base name of the output files (string, default:out)

varscan_perl_file the varscan (perl) executable (file)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.45 vmatch

A.45.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (vmatch)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

vmatch :

A.45.2 Parameters

Required parameters

vmatch_db vmatch db to compare against (file)

vmatch_input_file input file with the sequences to map (file)

title A job name - Describe what you are doing (string)

Optional parameters

vmatch_extra_parameters extra parameters to feed to vmatch (string)

vmatch_invert_gff Invert the GFF (T|*F*)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

A.46 vmatchdb

Builds a vmatchdb index from a sequence

A.46.1 Targets

(empty) Leaving the target unspecified executes the default target(s): (vmatchdb)

clean removes all results from this job

all executes the default target and into subdirectories to execute any other moa makefile it encounters

vmatchdb :

A.46.2 Parameters

Required parameters

vmatch_input_dir Input files for vmatch (directory)

vmatchdb_name Name of the vmatch index to create (string)

title A job name - Describe what you are doing (string)

Optional parameters

vmatch_input_extension file extension for the files in vmatch_input_dir (string, default:fasta)

vmatch_input_glob glob to select a subset of files from vmatch_input_dir (string, default:*)

vmatch_pl Prefix length (integer)

moa_precommand A single command to be executed before the main operation starts. For more complicated processing, please override the moa_preprocess target in the local Makefile. (string)

moa_postcommand A single shell command to be executed after the Moa is finished. For more complex processing please override the moa_postprocess target in the local Makefile. (string)

Bibliography

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J Mol Biol*, 215(3):403–410, Oct 1990.

[2] Biopython. http://biopython.org/.

[3] Generic Feature Format (v3). http://song.sourceforge.net/gff3.shtml.

[4] Gnu Make. http://www.gnu.org/software/make/.

[5] Red Hat Enterprise Linux. http://www.redhat.com/rhel/.

[6] P. Rice, I. Longden, and A. Bleasby. Emboss: the european molecular biology open software suite. *Trends Genet*, 16(6):276–277, Jun 2000.

[7] Lincoln D Stein, Christopher Mungall, ShengQiang Shu, Michael Caudy, Marco Mangone, Allen Day, Elizabeth Nickerson, Jason E Stajich, Todd W Harris, Adrian Arva, and Suzanna Lewis. The generic genome browser: a building block for a model organism system database. *Genome Res*, 12(10): 1599–1610, Oct 2002.

[8] Ubuntu. http://www.ubuntu.com/.