DOMINO

Development of Molecular Markers in Non-Model Organisms

Cristina Frías-López
José F. Sánchez-Herrero
Miquel A. Arnedo
Alejandro Sánchez-Gracia
Julio Rozas

Departament de Genètica, Microbiologia i Estadística
Departament Biologia Evolutiva, Ecologia i Ciències Ambientals
Institut de Recerca de la Biodiversitat (IRBio)

Universitat de Barcelona

http://www.ub.es/softevol/domino

May 10, 2016

UNIVERSITAT DE BARCELONA

EG&B Evolutionary Genomics & Bioinformatics

1 Overview

The development of molecular markers is one of the most important challenges in phylogenetic and genome wide population genetics studies, especially in non-model organisms. A highly promising approach for obtaining suitable markers is the utilization of genomic partitioning strategies for the simultaneous discovery and genotyping of a large number of markers. Unfortunately, some of these markers may not provide enough information to solve specific evolutionary questions.

We have developed DOMINO, a bioinformatics tool for informative marker development from both NGS data and pre-computed sequence alignments. The application implements popular NGS tools with new utilities in a highly versatile pipeline specifically designed to discover or select personalized markers at different levels of taxonomic resolution.

Availability and implementation: DOMINO is an open source and multiplatform software that uses Perl as the main scripting language for the new implemented functions and the Qt framework for the graphical user interface. The software is freely available from www.ub.edu/softevol/domino.

Authors

Cristina Frías-López	cristinafriaslopez@ub.edu
Jose Francisco Sánchez-Herrero	jfsanchezherrero@ub.edu
Miquel A. Arnedo	marnedo@ub.edu
Alejandro Sánchez-Gracia	elsanchez@ub.edu
Julio Rozas	jrozas@ub.edu

DOMINO Publication

Frías-López, C.*, Sánchez-Herrero, J. F.*, Guirao-Rico, S., Mora, E., Arnedo, M. A., Sánchez-Gracia, A. and Rozas, J. 2016. DOMINO: Development and selection of informative molecular markers for studies in non-model organisms. *In Preparation*.
*, equal contribution

DOMINO Web Site

www.ub.edu/softevol/domino

2 Installation

DOMINO is distributed as compressed archives, which includes all files needed to install/run the software (source code, executable binaries and example data files, from the DOMINO website (www.ub.edu/softevol/domino).

DOMINO Distribution Package

The distributed package contains the following files:

Installer and Pre-compiled versions

DOMINO_version_OSX_Installer.dmg. Installer for Mac OS X operating systems.

DOMINO_version_OSX_Compiled.zip. Pre-compiled version for Mac OS X operating systems.

DOMINO_version_LinuxDistribution_Installer.run. Installer for Linux distributions.

DOMINO_version_LinuxDistribution_Compiled.rar. Pre-compiled version for Linux distributions.

Source code, example files and documentation

In the DOMINO Web page we have also included the source code, some example files and the documentation of the program.

DOMINO_Repository.zip. Compressed folder including all files needed to compile the software:

- /**docs**. Folder that includes documentation and some miscellaneous information.

- /**example**. Folder with the 4 FASTQ files of the example data set.

- /**src**. Folder that includes the source code of DOMINO. The DOMINO_Qt_code folder includes the Qt and the C++ code, while the DOMINO_perl_code folder includes the new Perl scripts specifically developed for the project.

- /**files**. Folder with the DOMINO installation files, including the compressed files of Perl modules, executable binaries, and third party programs (already compiled). Some installers for Linux and Mac OS X systems are also included.

- **install.sh**: Shell script for installing the command-line version of DOMINO.

- **README and NEWS**. Text files with relevant information about DOMINO.

- **Change.log.** Text file containing the different updates of the DOMINO project.

- **LICENSE.txt. FALTA ESTO**

DOMINO Software Requirements

Mac OS Systems (Mac OSX 10.x.x or higher)

Perl programming language. Perl should be installed by default on Mac OS X operating systems. If not, please follow instructions at http://learn.perl.org/installing/osx.html.

C++ compiler. This compiler is included in the Xcode integrated development environment, which can be installed via App Store.

`zlib`. This compression library should be installed by default on Mac OS X operating systems.

Linux systems (tested in Ubuntu 10.04 LTS)

`Perl` programming language. Perl should be installed by default on Linux distributions. If not, please follow instructions at http://learn.perl.org/installing/unix_linux.html.

C++ compiler. This compiler is included in the build essential package or can be installed thought the system package management tool (`g++` compiler).

Ubuntu: `sudo apt-get install build-essential`

`zlib`. This compression library can be installed thought the system package management tool. It is also available at http://www.zlib.net/

Ubuntu: `sudo apt-get install zlib1g-dev`

Windows Systems

In preparation

DOMINO Installation

DOMINO complete version

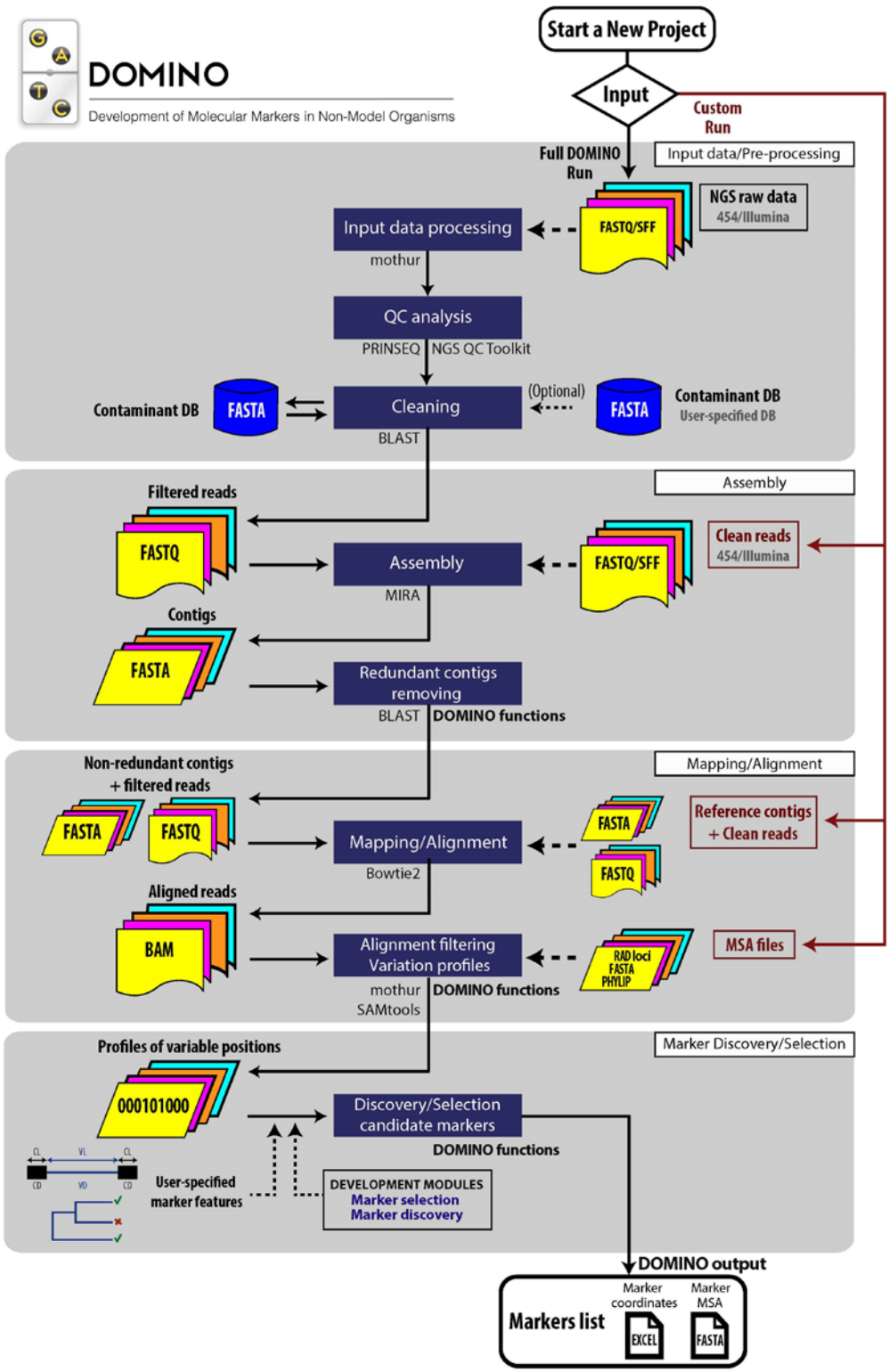
By executing the installers included in the DOMINO distribution package (`DOMINO_version_OSX_Installer.dmg/DOMINO_version_LinuxDistribution Installer.run`), all needed steps to install the software, including the creation of necessary folders and files as well as the Desktop icon, will be completed. After the installation, DOMINO can be run either under the GUI or under the command-line version.

DOMINO command-line version

In the command-line prompt (shell) run `sh install.sh` and follow the instructions. The installer will generate all folders and files necessary to run DOMINO under the command line version in Mac and Linux systems. All DM `Perl` scripts as well as the executable files of the external software integrated in the pipeline (third-party software) will be included within the directory `/bin`.

3 DOMINO –Workflow & Quick Guide

The DOMINO GUI consists in six TABs (GUI windows) that must be successively applied (under the Full DOMINO Run), or selectively skipped (under the different Custom Run options). The workflow behind schematizes the principal running steps across the four main phases.



1) Start a New Project

This TAB provides DOMINO information on the location of the relevant directories: the `Perl` and the DOMINO directories as well as other general options.

2) Input Data / Pre-Processing

Use this TAB to input your NGS data files. These files must contain long (454) or short (Illumina single or paired-end) raw data typically from a reduced partitioning protocol with several taxa (taxa panel). The sequences of each taxon should be properly barcoded (aka, tag, MID or index) or entered in separate files. In this phase**,** DOMINO will conduct all needed read pre-processing steps (i.e., cleaning and trimming of low quality, low complexity, short length and contaminant reads). In this window the user can configure some of the pre-processing parameters.

Scripts and software: `Mothur`, `PRINSEQ`, `NGS QC Toolkit`, `BLAST`, in-house `Perl` scripts.
Input: Raw reads
Output: Pre-processed reads

3) Assembly

This TAB allows setting some of the parameters for the *de novo* assembly phase, which will be performed separately for each taxon. After the assembly, DOMINO will remove repetitive contigs and reads for further steps.

Scripts and software: `MIRA`, `BLAST`, in-house `DOMINO Perl` scripts (`DM` scripts).
Input: Pre-processed reads
Output: Non-redundant contigs and unassembled reads

4) Mapping / Alignment

This TAB allows specifying the relevant parameters to obtain the profiles of variable sites between pairs of taxa (pairwise profiles). Here, the user can also enter their own pre-processed reads, reads plus reference sequences or multiple sequence alignments (MSA).

DOMINO will map the pre-processed reads to the contigs from the assembly phase or to the supplied reference sequence(s) and will perform the post-mapping quality-filtering and a conservative variant calling.

Scripts and software: `Bowtie2`, `Mothur`, `SAMtools`, and `DM` scripts.
Input: Pre-processed reads and reference sequence(s); MSA in various formats (see below for details).
Output: Profiles of variable sites across taxa

5) Marker Discovery / Selection

This TAB allows the user to select the taxa and the parameters values to be used for discovering or selecting the desired informative markers.

Scripts and software: `DM` scripts.
Input: Profiles of variable sites across taxa
Output: list of designed/selected informative markers

6) Files Viewer

This TAB shows the relevant files and directories used in each DOMINO run, and the list of the informative markers identified or selected with their coordinates (in contigs or reference sequence(s)).

EXAMPLE DATA FILE

To familiarise the user with the DOMINO GUI, we have included in the distributed package an example dataset (`4000Nemesia_fastq_files.zip`), of a panel with four taxa (which is a small subset of the RRL reported in Frías-López *et al.* 2016). This data set is used throughout this

manual to illustrate the different phases of the DOMINO workflow. It includes raw data (a subset of 4,000 reads from a 454 sequencing of an RLL experiment; 2.1 Mbp total; average read length of 518 bp; N50 of 611 bp) in four FASTQ files (one FASTQ file per taxon: `N_raripilia061.fastq`; `N_raripilia079.fastq`; `Nemesia_sp043.fastq`; `I_brauni098.fastq`).

Computational requirements issues

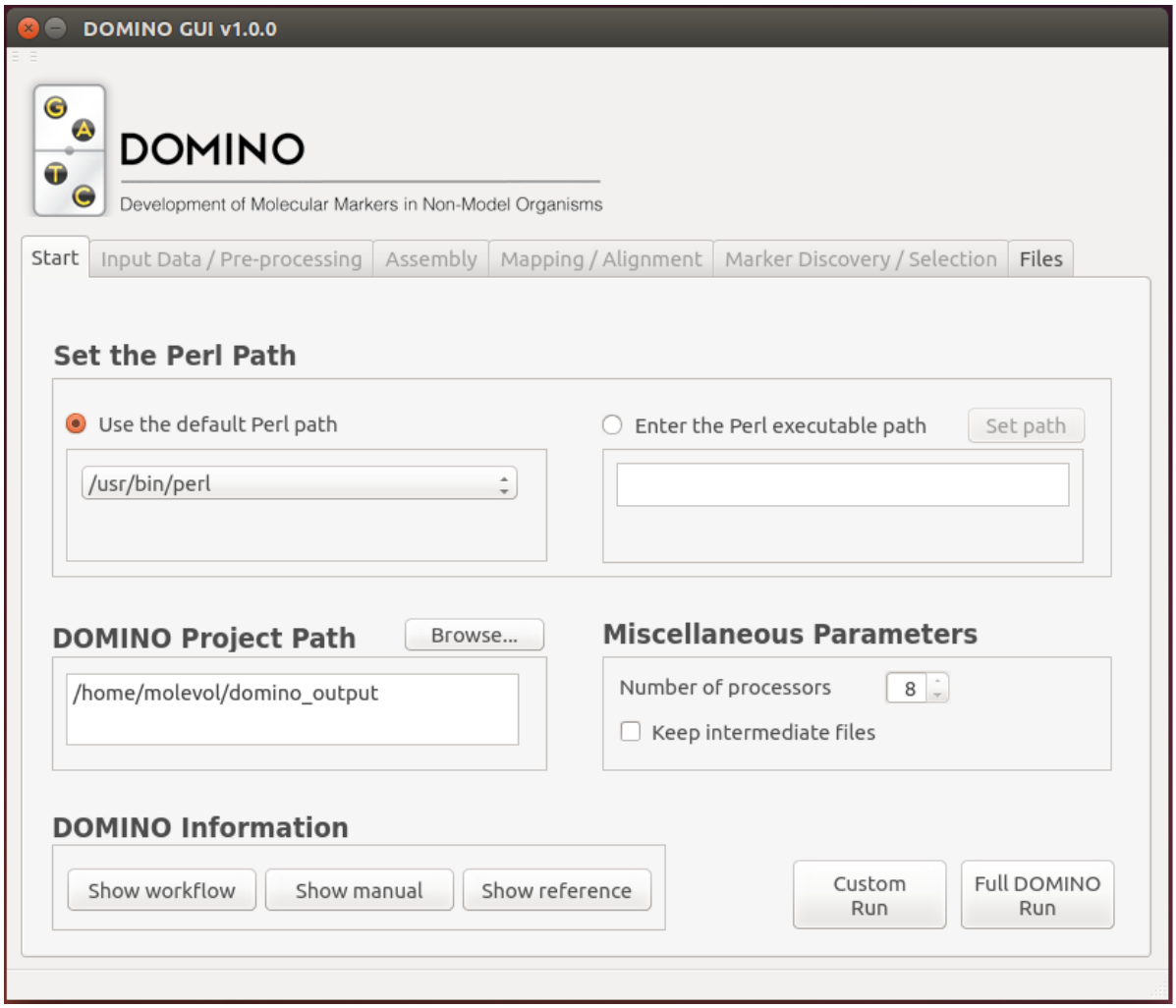
An important point to be considered before using DOMINO with NGS data (especially in the Pre-processing and Assembly phases) is the raw data size. Although the bioinformatics tools included in DOMINO can perfectly handle these kinds of data, they can consume substantial amounts of computational resources, especially RAM memory. We do not recommend applying the Full DOMINIO run with data sets of many millions of short reads in a typical desktop computer (with an insufficient number of CPUs and limited RAM); this process might take a very long (unacceptable) time to complete, or even cause a system crash.

For massive NGS data sets (typically more than 5 million short reads per file), the user can either run DOMINO under the command line version using high performance computers (i.e., a computer cluster with high large amounts of CPUs and lots of RAM and hard disk space) or, take advantage of the GUI Custom run options and enter DOMINO partially processed data, e.g. pre-processed reads, pre-assembled contigs or alignment files (SAM/BAM) obtained from other more memory-efficient software.

Using the command-line option

The user can also run DOMINO under the command-line prompt. This option allows running domino in high performance computer clusters and managing some extra options and parameters than the GUI version, such as a second (optional) iterative assembly round with CAP3 (using the contigs and singletons obtained from MIRA as input data), or activate extended options in the Alignment/Mapping and Marker Discovery/Selection phases. See the DOMINO website to find more information of the basic DOMINO command-line option, as well as for the default parameter values used in the GUI.

3 DOMINO --START Your Project



Set the Perl and DOMINO Path boxes

In this first TAB, the user should specify the path to the folder with the `Perl` scripting language, either selecting the default `Perl` executable file (Default Perl path, usually recognized in all UNIX-based systems) or entering a different path name (Enter Perl executable path). The DOMINO path project, which will be used to write the output files, can also be specified here.

Miscellaneous Parameters box

In this box, the user is asked for the number of processors (cores) to be used for computation and whether the intermediate files will be kept or eliminated.

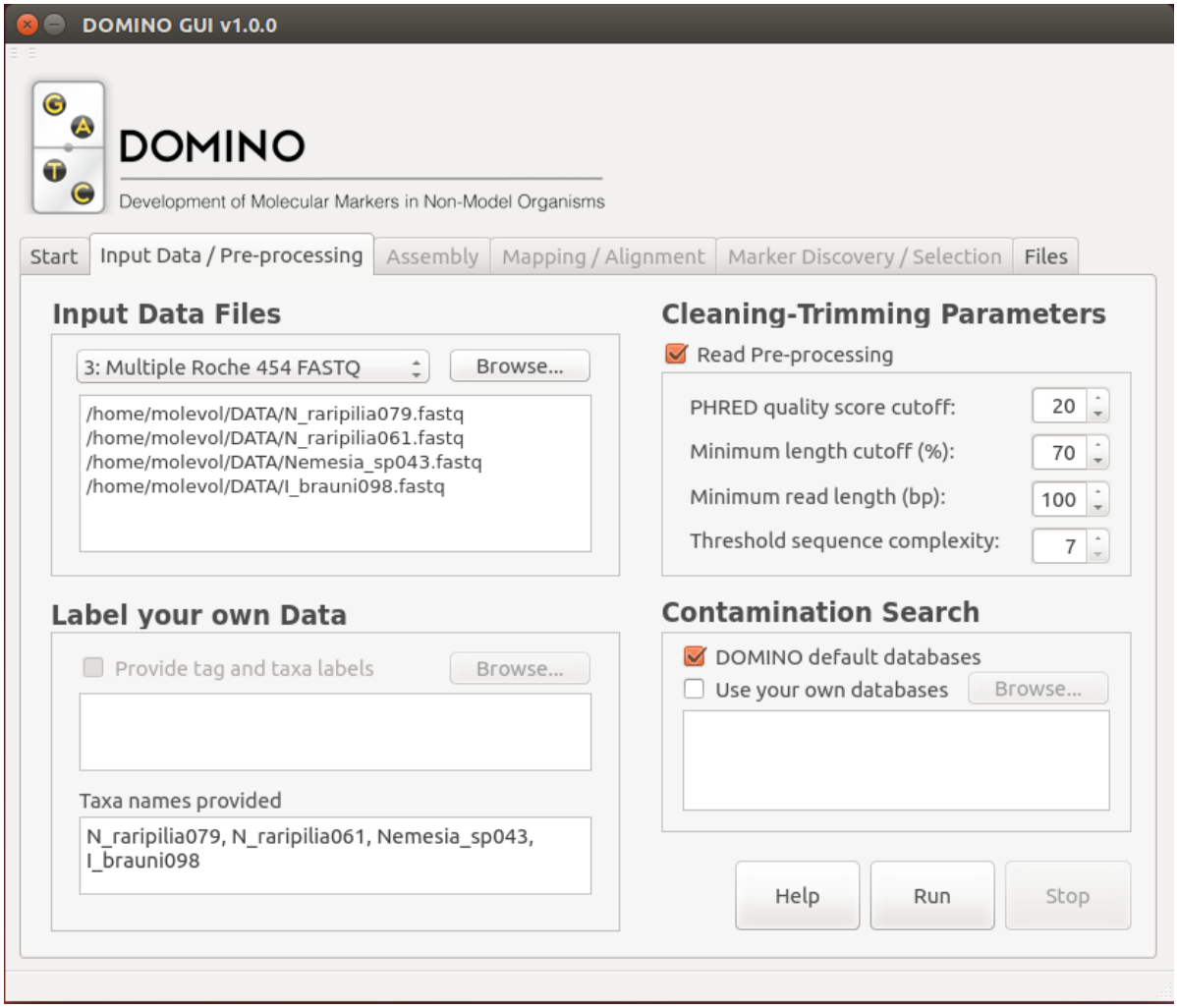
Full DOMINO Run button

This button starts a complete DOMINO run (that is, performing consecutively all core steps of the four DOMINO phases), from an input with raw NGS data to the final listing of candidate markers.

Custom Run button

Use this button to skip some DOMINO phases. Under this option, users can load their own cleaned reads, reference sequence(s) or pre-computed MSA. This option also permits resume a previous DOMINO execution.

4 DOMINO --Input Data/Pre-Processing



Input Data box

DOMINO accepts different types of NGS input data files. The program can handle 454-SFF files and FASTQ files with 454, Illumina single or paired-ends raw reads (in various formats). The raw data of each taxon can be entered separately, or combined in a single file (such as in many SFF files). In this case, DOMINO requires that the DNA sequence data of each taxon was appropriately labelled (tagged with barcodes -MIDs).

Note that in a full run from short reads, DOMINO applies an assembly-based approach; the program is therefore optimized to work with genome partitioning methods in which the length of the size-selected or enriched fragments and the sequencing depth are enough to permit the assembly of putative homologous fragments. For data from other sequencing approaches (RAD-based data, such as. RAD-Seq, ddRAD or GBS) see the Mapping/Alignment and Marker Discovery/Selection sections of this manual.

Accepted data types

1: Roche 454 SFF

A single file in the Standard Flowgram Format (SFF), containing all 454 reads of all taxa from the panel, accordingly tagged (with MID tags). File Extension: `*.sff`

2: Roche 454 FASTQ
A single file in FASTQ format, containing the 454 reads of all taxa from the panel together, accordingly tagged (with MID tags). File Extension: `*.fastq`

3: Multiple Roche 454 FASTQ
Multiple FASTQ files, each file should contain the 454 reads of each taxon from the panel. File Extension: `*.fastq`

4: Illumina single-end FASTQ
A single file in FASTQ format, containing Illumina single-end reads of all taxa from the panel, accordingly tagged (with MID tags). File Extension: `*.fastq`

5: Multiple Illumina single-end FASTQ
Multiple single-end FASTQ files, each file should contain the raw illumina single-end reads of each taxon from the panel. File Extension: `*.fastq`

6: Two Illumina paired-end FASTQ
Two FASTQ files, one file should contain the left "_R1", and the other the right "_R2" fragment ends from a paired-end Illumina experiment, sequencing all taxa from the panel. Each taxon should be appropriately tagged (with MID tags). File Extension: `*.fastq`

7: Multiple Illumina paired-end FASTQ
Multiple paired-end FASTQ files. In this case the user should provide two illumina files of each taxon from the panel, one for the left "_R1" and another for the right "_R2" fragment ends. File Extension: `*.fastq`

Filenames
Structure:
`File_name[_Rn].extension`, where:
`File_name` stands for the taxon identifier (periods, commas or blank spaces are not allowed).
`[_Rn]` is optional and it is used to indicate the left "_R1" or right "_R2" fragment ends of a paired-end sequencing experiment.
`extension` stands for data type (SFF or FASTQ files).

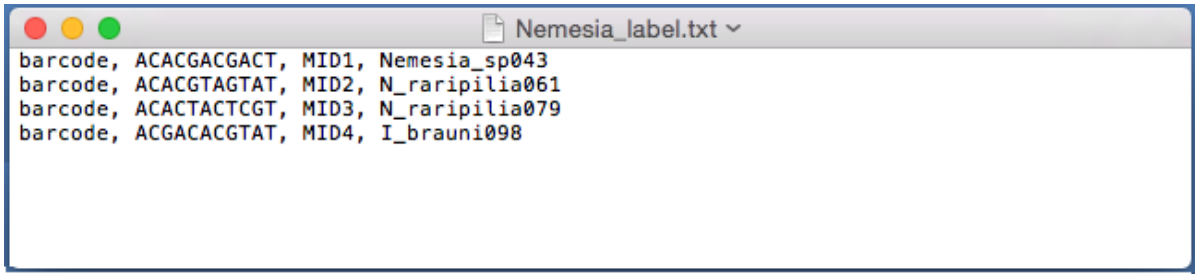
Examples:
`I_brauni098.fastq`, a FASTQ file with single-end reads from the *I_brauni098* taxon.
`Dmelanogaster_R1.fastq`, a FASTQ file with the left reads (paired-end data) of *Dmelanogaster* taxon.
`Dmelanogaster.fastq`, a FASTQ file with reads of *Dmelanogaster* taxon.

Taxon names
Taxon name lengths must be less than 25 characters. In case of larger filenames, you can rename your files including the tag `id-` to indicate the part of the filename that DOMINO will use as taxon name.

Example:
`FileXXX236363663id-Dmelanogaster_R1.fastq` --> taxon name: *Dmelanogaster*
`File1234-id-Dmelanogaster.fastq` --> taxon name: *Dmelanogaster*

Label your Data box

Since DOMINO accepts different types of input data files, it might require extra information in some cases. When the input file is a single 454 SFF or 454/Illumina FASTQ file, the program will request information on the particular nucleotide sequence used as MIDs or, alternatively, the names of the taxa in the panel (Provide a Tag & Taxa labels option). The user must provide this information in a text file (see the example below; file: `Nemesia_label.txt` the DOMINO website.



Loading separate input data files (e.g., multiple FASTQ files), each one with information from a single taxon, DOMINO will use the left part of the filename as the taxon name, excluding, if any, the "_R1" or "_R2" labels, which are used to indicate the left or right reads of a paired-end. For instance, taxon name extracted from the filename `I_brauni098.fastq`, will be *I_brauni098*, while `Dmelanogaster_R1.fastq`, `Dmelanogaster_R2.fastq` and `Dmelanogaster.fastq` will have the same specific taxon name (*Dmelanogaster*).

Cleaning-Trimming parameters box

Use this box to set the values for read pre-processing parameters. Nucleotides with quality values lower than the specified *PHRED quality score cut-off*, the reads with a % of valid nucleotides lower than the selected Minimum length cut-off, the reads shorter than the stablished Minimum read length, and the reads with Threshold sequence complexity values lower than those pre-defined by the user, will not be used in further steps.

Contamination Search box

This box allows selecting the database for the contaminants filtering step. DOMINO includes the UniVec database (which include information of vectors, adapters, linkers or other cloning contaminants), and the genome sequence of some prokaryotic (including *E. coli*) species and some virus as default databases for performing this task. The user can also load their preferred database (Use my own databases option) in FASTA format (see the file `MyOwnContaminantDB.txt` in the DOMINO website as an example).

By default DOMINO databases include the following prokaryotic species databases:
* *Escherichia coli* BL21(DE3) chromosome, complete genome
gi: 387825439; ref: NC_012971.2

* *Pseudomonas aeruginosa* M18 chromosome, complete genome
gi: 386056071; ref: NC_017548.1

* *Saccharomyces cerevisiae* S288c chromosome I, complete sequence
gi: 330443391; ref: NC_001133.9

* *Staphylococcus aureus* subsp. aureus 6850, complete genome
gi: 537441500; ref: NC_022222.1

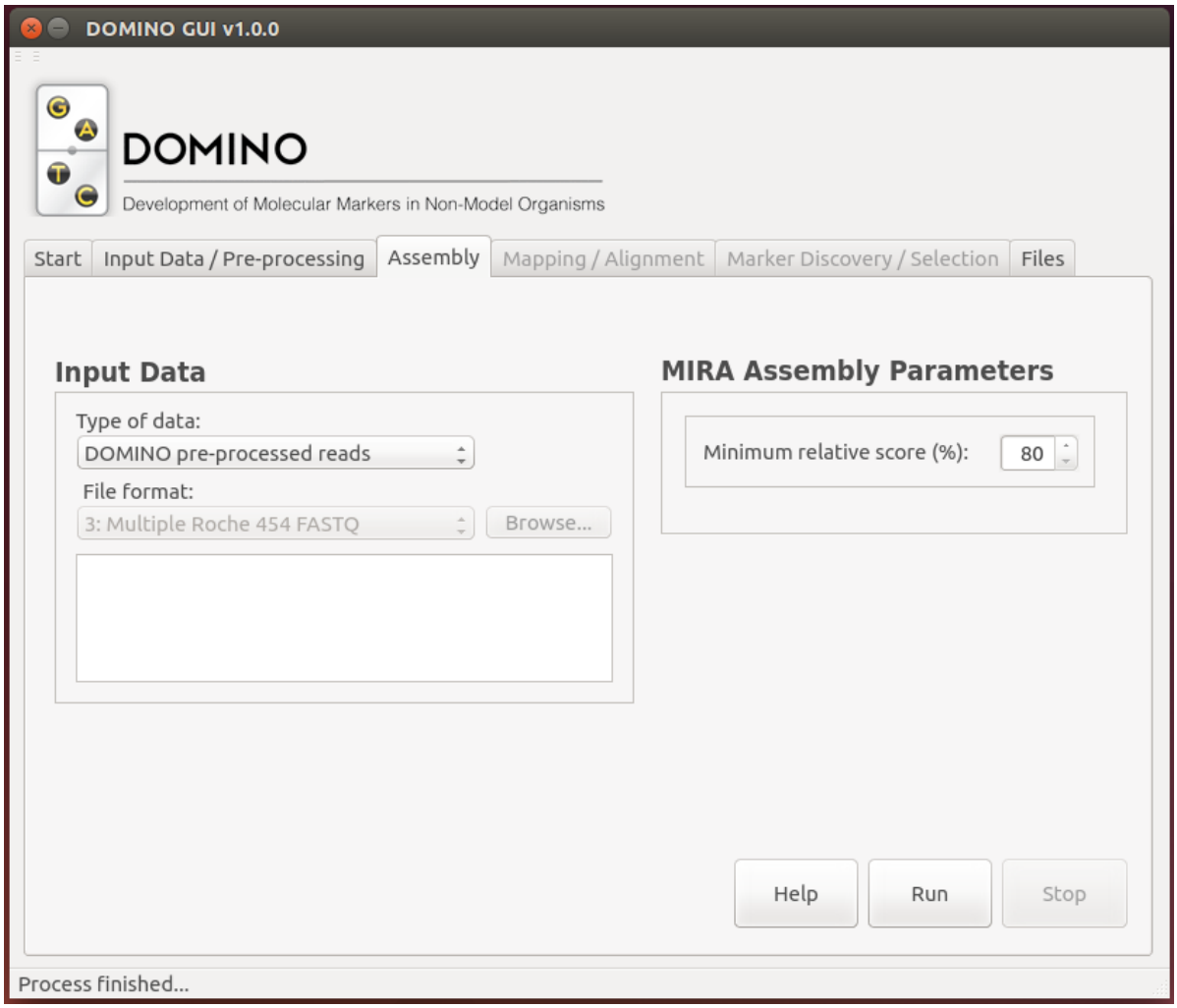
Results of our example data file

Using the default parameter values on the example data set (4,000 raw reads), DOMINO will select a final set of 3,981 high quality pre-processed reads.

Additional Information

The pre-processing steps are performed by using several scripts and software, such as `Mothur`, `PRINSEQ`, `NGS QC Toolkit`, `Blast` and several new functions implemented in an in-house written `Perl` script (`DM_Cleaning.pl` PERL script). Please read the documentation provided by these software and their description in the DOMINO paper (Frías-López *et al.* 2016) for details on parameters and options. See the DOMINO website to see some examples of the input file formats accepted by DOMINO.

5 DOMINO --Assembly



DOMINO performs the *de novo* assembly of pre-processed reads separately for each taxon from the panel by using MIRA. DOMINO identifies the reads encompassing repetitive regions (reads classified as HAF6, HAF7 and MNRr in MIRA; see the MIRA documentation for details), and remove them from further DOMINO steps. Later, and to avoid including redundant contigs in the next steps, DOMINO conducts an all-vs-all contigs BLAST search. Contigs involved in positive hits with low e-values (cut-of E-value of 10^{-50}), with an overlapped region higher or equal than 85% (of the shorter contig length), and minimum similarity value of 85% (of the overlapped region) are collapsed, i. e., only the longest contig of each pair in a positive Blast hit will be used as a reference sequence for the next Mapping/Alignment step.

Input Data box

In this box, the reads pre-processed in previous DOMINO executions (in a complete/standard Full DOMINO Run option) or the reads directly supplied by the user (under the Custom Run option) can be loaded to perform the assembly.

Type of data: Data types

There are two options:

DOMINO pre-processed reads

This is the default option. Use this option if you want to perform the assembly using the reads previously pre-processed by DOMINO (either in the current or in a previous DOMINO execution).

User-supplied pre-processed reads
Using the Custom Run option, the user can enter directly in the assembly phase, by skipping all previous pre-processing steps. In this case, the user must supply its own data files in one of the accepted formats through the *File format* option. The filename and taxon name structure must be as specified in the Filenames and Taxon names sections in the description of previous TAB. In order to avoid problems with read naming and pair-end nomenclature we strongly recommend to make use of the Input Data/Pre-processing TAB to prepare user-supplied input files for the DOMIMO Assembly phase (i.e., by running a DOMINO pre-processing step in the previous TAB with the cleaning-trimming and contamination search options deselected).

File format: Accepted data types
3: Multiple Roche 454 FASTQ
Multiple FASTQ files, each file should contain the raw 454 reads of each taxon from the panel. File Extension: `*.fastq`

5: Multiple Illumina single-end FASTQ
Multiple single-end FASTQ files, each file should contain the raw illumina single-end reads of each taxon from the panel. File Extension: `*.fastq`

7: Multiple Illumina paired-end FASTQ
Multiple paired-end FASTQ files. In this case the user should provide two illumina files of each taxon from the panel, one for the left "_R1" and another for the right "_R2" fragment ends. File Extension: `*.fastq`

The filename structure is given in the previous TAB (Input Data/Pre-processing TAB).

MIRA Assembly Parameters
In this box, the user can set the values of some relevant parameters for the *de novo* assembly phase with MIRA, such as the minimum % matching for the assembly of two reads (Minimum relative score). In case of a second (optional) CAP3 assembly, the user can specify the minimum overlapping length and the minimum % of identity accepted in these overlapping regions.
*Note***.** The default values for these parameters has been set accordingly with the type of data (e.g. they are different for 454 and Illumina data).

Results of our example data file
The MIRA assembly, using the default DOMINO values, of each of the four taxon in the example, will generate 100, 128, 134 and 145 contigs for *N_raripilia061*, *N_raripilia079*, *Nemesia_sp043* and *I_brauni098*, respectively (see also the Frías-López *et al.* 2016; supplementary Tables S4-S5).

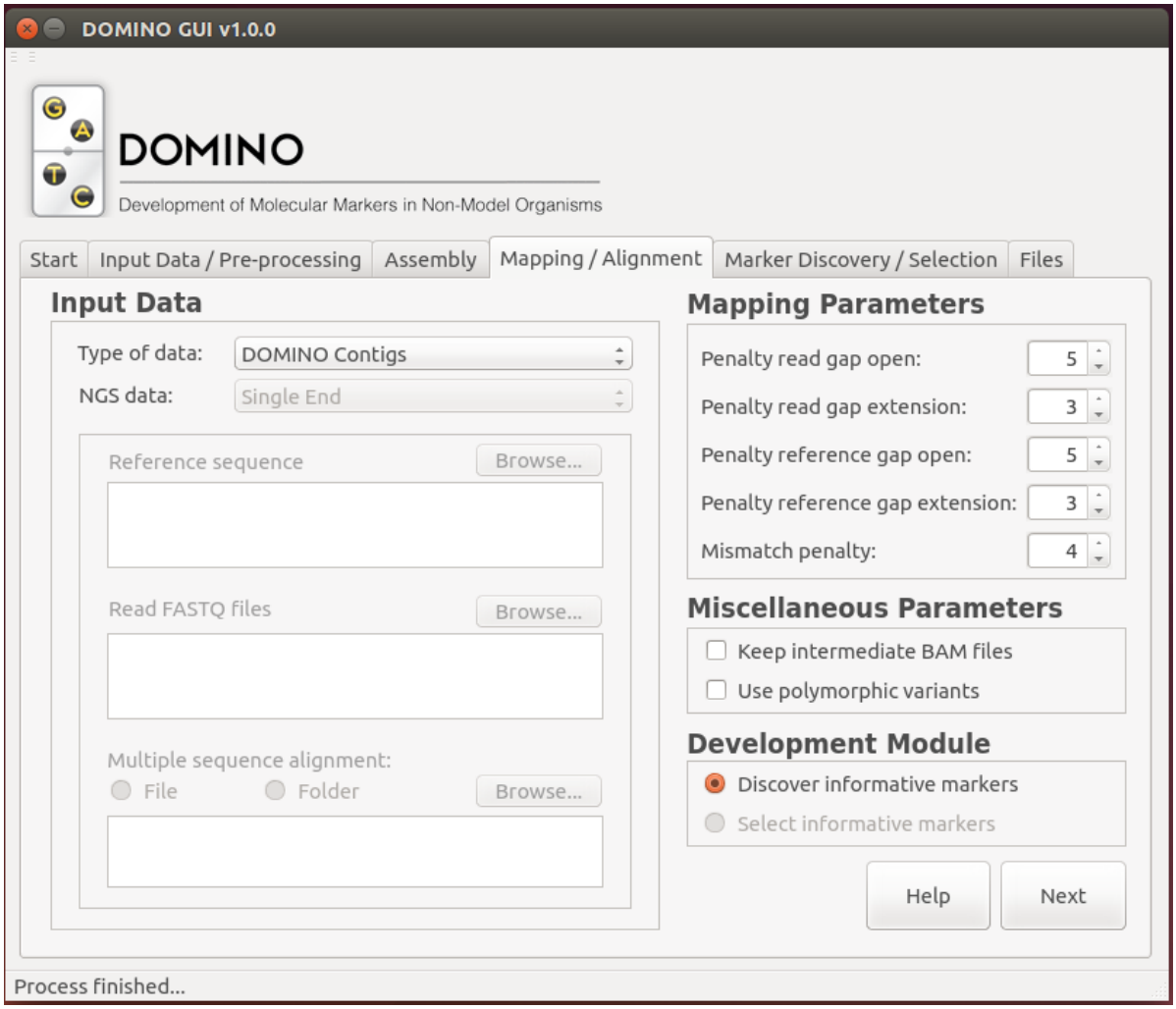
MIRA limitations

MIRA is based on a highly accurate overlap graph algorithm and usually shows good performance with short reads and contigs, as those expected to be handled by DOMINO (data from typical genome partitioning, low coverage or small genome size experiments). For the assembly of much larger sequences (which needs many millions of short reads) the overlap graph strategy becomes computationally intensive and should be either avoided or carried out in high performance computers (see the Computational requirements issues section7).

Additional Information

In addition to MIRA (and CAP3) software, DOMINO, uses a series of new developed functions implemented in an in-house `Perl` script (`DM_Assembly.pl` PERL script) for the assembly phase. Please read the documentation provided by these external software and their description in the DOMINO paper (Frías-López *et al.* 2016) for details on parameters and options. See the DOMINO website to see some examples of the input file formats accepted by DOMINO.

6 DOMINO ––Mapping/Alignment



At the end of Mapping/Alignment phase, DOMINO builds the arrays of variable positions between pairs of taxa required for further marker Discovery. This phase is performed in four different steps.

Mapping/Alignment
First, the pre-processed reads from each taxon are independently mapped back to the contigs (and singletons) obtained in the assembly phase using `Bowtie2`. The resulting BAM files [*n* x (*n*-1) files, *n* = number of taxa in the panel; four in our example] will contain all pairwise combinations of reads from one taxon aligned back to the contigs of all other taxa separately (i.e., ignoring the alignments between reads and contigs from the same taxon).

For instance, using the assembly reference sequence data of taxon#1, DOMINO will create 3 different BAM files (contigs of taxon#1 with reads of taxon#2, with reads of taxon#3, and with reads of taxon#4), and so on. Using the example dataset (a panel of four taxa), DOMINO will perform four separated assemblies, one per taxon; in the mapping step, the pre-processed reads of each of these four taxa will be aligned separately upon contigs of the other three taxa, that is, a total of 4 x 3 = 12 BAM/SAM files.

Filtering mapping errors & non-useful alignments
DOMINO applies some additional filtering steps to remove alignments with mapping errors, unmapped regions or multimapping reads. These problematic alignments can definitely generate

false informative markers (regions with artefactual levels of nucleotide diversity). Alignments with an unusually large number of mapped reads, which might result from repetitive regions, are also removed since they are not useful as sources of informative markers. For that, DOMINO estimates the average read coverage (*c*) among all alignments, and removes those with coverage equal or greater than a critical value ($P < 10^{-5}$; this value could be modified in the command-line version), which is obtained from a Poisson distribution with mean *c*. After these filters DOMINO builds a pileup file for each of the filtered BAM files (12 in the example data set), using the SAMtools suite.

Filtering sequencing errors & ambiguity codes
Since sequencing errors can severely affect marker identification and selection, DOMINO implements a very conservative variant calling function (see Frías-López *et al.* 2016 supplemental methods for details) for detecting and masking putative errors. First, to avoid the calling of spurious nucleotide variants in low sequencing coverage experiments (i.e. erroneously assigned variants fixed between the taxa panel), DOMINO masks the information from positions with only one read mapped to the reference.

DOMINO incorporates a similar conservative criterion to use only highly credible polymorphisms when the Polymorphic variants option is activated (see the Marker Discovery/Selection section). For positions with 8 or more reads mapped, DOMINO discards those polymorphic variants in which the frequency of the minor allele is significantly lower than the expected for a diploid individual ($P < 0.05$ in Binomial distribution with $p = 0.5$), likely corresponding to a sequencing error. For lower coverage values, DOMINO will use the information of a polymorphic variant only if the frequency of the minor allele is present in two or more mapped reads.

DOMINO also checks the presence of positions with ambiguity codes (generated by MIRA) and decides whether they are variable depending on the nucleotides present in the rest of positions. For instance, if in the reference sequence appears a "Y" (IUPAC ambiguity code for "C/T"), and all nucleotides in the mapped reads have an "A" for this position, DOMINO considers this position as variable. On the other side, if all reads have a "C" in this position, DOMINO considers this position as invariable.

Profile of variable sites between pairs of taxa
At the end of the mapping step, DOMINO builds a single profile of variable sites (merged profile), combining the information from all pairwise pileup having the same reference sequence. In the example data set, DOMINO will build four merged profiles (one for each assembled taxon). In case that user entered a single reference sequence, DOMINO generates only one merged profile.

Input Data box

The mapping step can be performed using the DOMINO files from the current project, or by loading directly the data files necessary to conduct the mapping step (under the Custom Run option). Users can also input their own pre-processed reads, appropriately pre-processed accordingly with the genome partitioning methodology used to generate the library and the NGS technology used for sequencing, and MSA files. In this case, and in order to avoid problems with read naming and pair-end nomenclature, we strongly recommend to make use of the Input Data/Pre-processing TAB to prepare user-supplied input files for the DOMIMO Mapping/Alignment phase (i.e., by running a DOMINO pre-processing step in the previous TAB with the cleaning-trimming and contamination search options deselected).

Type of data: Data types
DOMINO accepts different types of data files:

DOMINO contigs (Full DOMINO Run option)
Default value. Use this option if you want to use the contigs previously assembled in DOMINO, either in the current or in a previous DOMINO execution.

Multiple taxa references (Custom Run option)
Use this option to map the loaded pre-processed reads to reference sequences from several taxa (a separate reference sequence per taxon). The user must load two different kind of data files:

1) Reference sequences (such as size-selected or enriched library fragments or genome contigs and scaffolds) from multiple taxa. The user must upload *n* data files (being *n* the number of taxa in the panel) in multi-FASTA format, with the DNA sequence to be used as a reference in the mapping/alignment phase.
2) NGS data files. FASTQ files (one file per taxon for single-end reads; two files for paired-end reads) with the sequence of the pre-processed reads to be aligned to the supplied reference sequences. DOMINO will therefore conduct *n*(*n*-1) mapping/alignment steps. In addition, the user must indicate whether the supplied reads are from a single-end or paired-end sequencing experiment.

Single taxon reference(s) (Custom Run option)
Use this option to map the loaded pre-processed reads from one or more taxa to a single taxon reference sequence. This option is identical to the References from multiple taxa option, but using only a single reference sequence for all taxa. The user must load two kind of data files:

1) A reference sequence (such as size-selected or enriched library fragments or genome contigs and scaffolds) from a single taxon. The user must upload one data file (in multi-FASTA format) with the DNA sequence to be used as a reference in the mapping/alignment phase.
2) NGS data files. FASTQ files (one file per taxon for single-end reads; two files for paired-end reads) with the sequence of the pre-processed reads to be aligned to the supplied reference sequence. DOMINO will conduct *n* mapping steps, being *n* the number of taxa in the panel. In addition the user must indicate whether the supplied reads are from a single-end or paired-end sequencing experiment.

MSA file(s) (Custom Run option)
Use this option to identify informative molecular markers directly in one or more MSA of nucleotide sequences; each MSA may include DNA sequence information from size-selected library fragments (from a genome partitioning scheme) or from genomic contigs or scaffolds. Using this option, DOMINO will skip the mapping phase and use directly these MSA for the next step (Marker Discovery/Selection TAB). The user can load the MSA in any of the following formats.

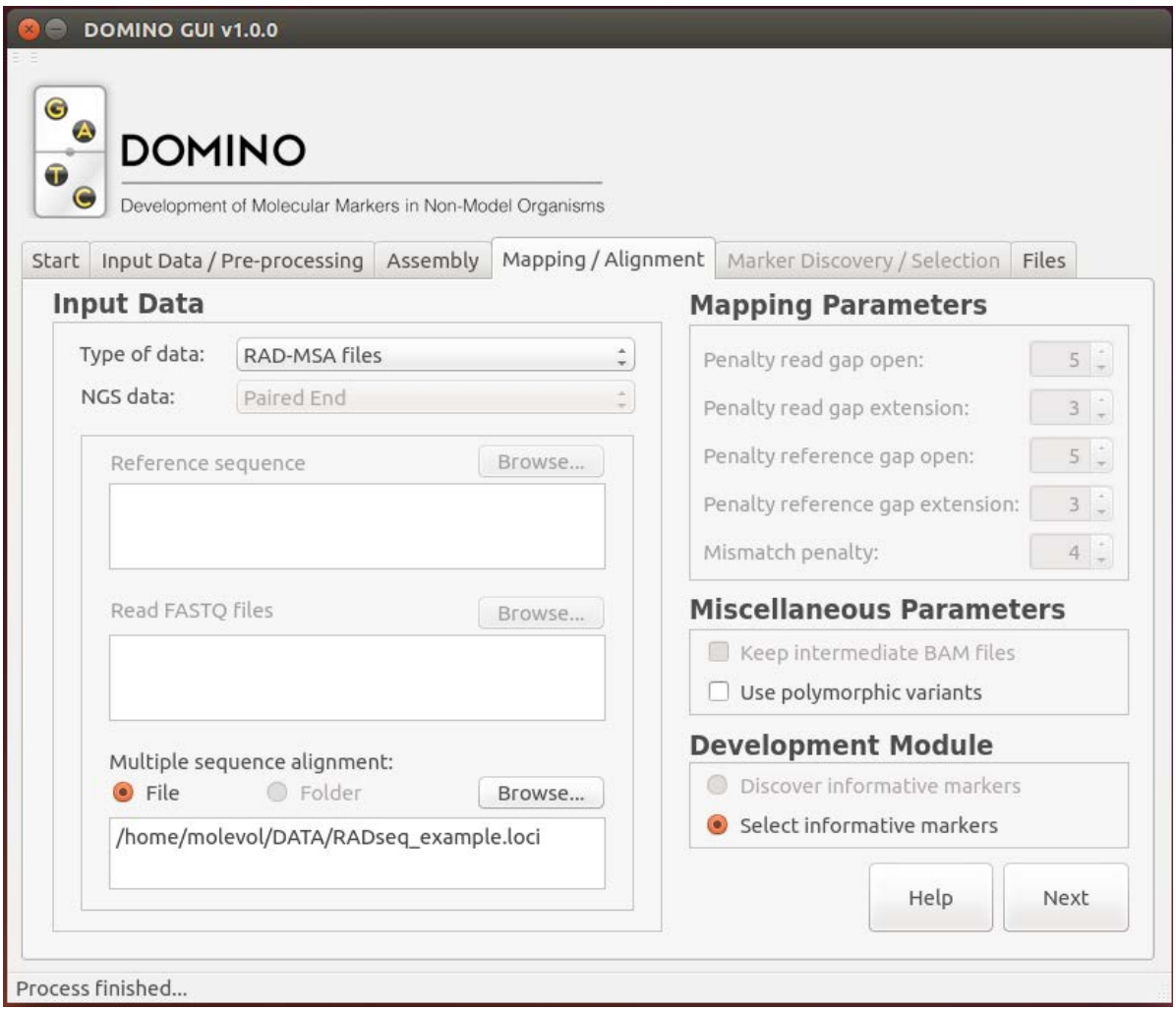
1) Various MSA files in PHYLIP or FASTA format. Each file must include only one MSA corresponding to a single library fragment or genomic region.
2) A multi-MSA file in PHYLIP format. A single data file with multiple MSA, each one in PHYLIP format. Each region are separated by the standard first-line PHYLIP identifiers (two numbers: the number of taxa, and the number of nucleotides).

RAD-MSA files (Custom Run option)
Use this option to select the most informative MSAs (RAD loci) among a supplied set, which must to be previously generated by some RAD software tools (e.g. PyRAD or Stacks software for RAD-Seq analyses). Using this option, DOMINO will skip the mapping phase and use directly these MSA for the next step (Marker Discovery/Selection TAB). The user can load the MSA in any of the following formats:

1) A multi-MSA file in PyRAD output loci format (the output file from PyRAD with the extension `*.loci`; see the PyRAD documentation for details). A single data file with all

MSA (one per RAD loci) from a RAD-Seq or a similar methodology. Each MSA must to be in FASTA format and separated by the character: `//`.

2) A multi-MSA file in Stacks output FASTA format (the output file from Stacks with the name `batch_X.fa`; see the Stacks documentation for details). A single data file with all MSA (one per RAD loci) from a RAD-Seq or a similar methodology. The sequence of the two haplotypes of each individual is included in each MSA.



Filenames and taxon names

The data files for the same taxa should have the same taxon id-name. The accepted filename structures are:

`[xx]id-yyy.fastq` --> Clean reads (FASTQ format) for taxon *yyy*

`[xx]id-yyy[_Rn].fastq` --> Clean paired-end FASTQ reads for taxon *yyy*. _Rn, indicate the left "_R1" or right "_R2" reads of a paired-end sequencing experiment

`[xx]id-yyy.contigs.fasta` --> Contigs for taxon *yyy* (FASTA format)

`[xx]id-yyy.fasta` --> Reference sequence (scaffolds; complete genome) for a particular taxon of the panel (FASTA format)

Where:

`xx`, could be any character or none

`yyy`, is the desired taxa name.

Therefore, some correct filenames could be:

`id-HomoSapiens.fastq`

```
123id-Dmelanogaster_R1.fastq
1id-Nemesia.contigs.fasta
Myid-Buthus.fasta
```

The accepted filenames structure and extension for the MSA of RAD loci are:
`Filename.loci` --> Multi-MSA in PyRAD loci output format
`Filename.fa` --> Multi-MSA in Stacks FASTA output format

Mapping Parameters (Bowtie2 software) box

In this box, the user can modify penalty parameters for the Bowtie2 mapping. Please read Bowtie2 documentation.

Miscellaneous Parameters

Skip DOMINO mapping step. This box allows skipping the mapping step, and use mapping data from a previous DOMINO session.
Skip DOMINO MSA-parsing step. This box allows skipping the generation of variation profiles, and use profiles data from a previous DOMINO session.

Development Module

Use this box to select the specific DOMINO module to be used in the Discovery/Selection phase.

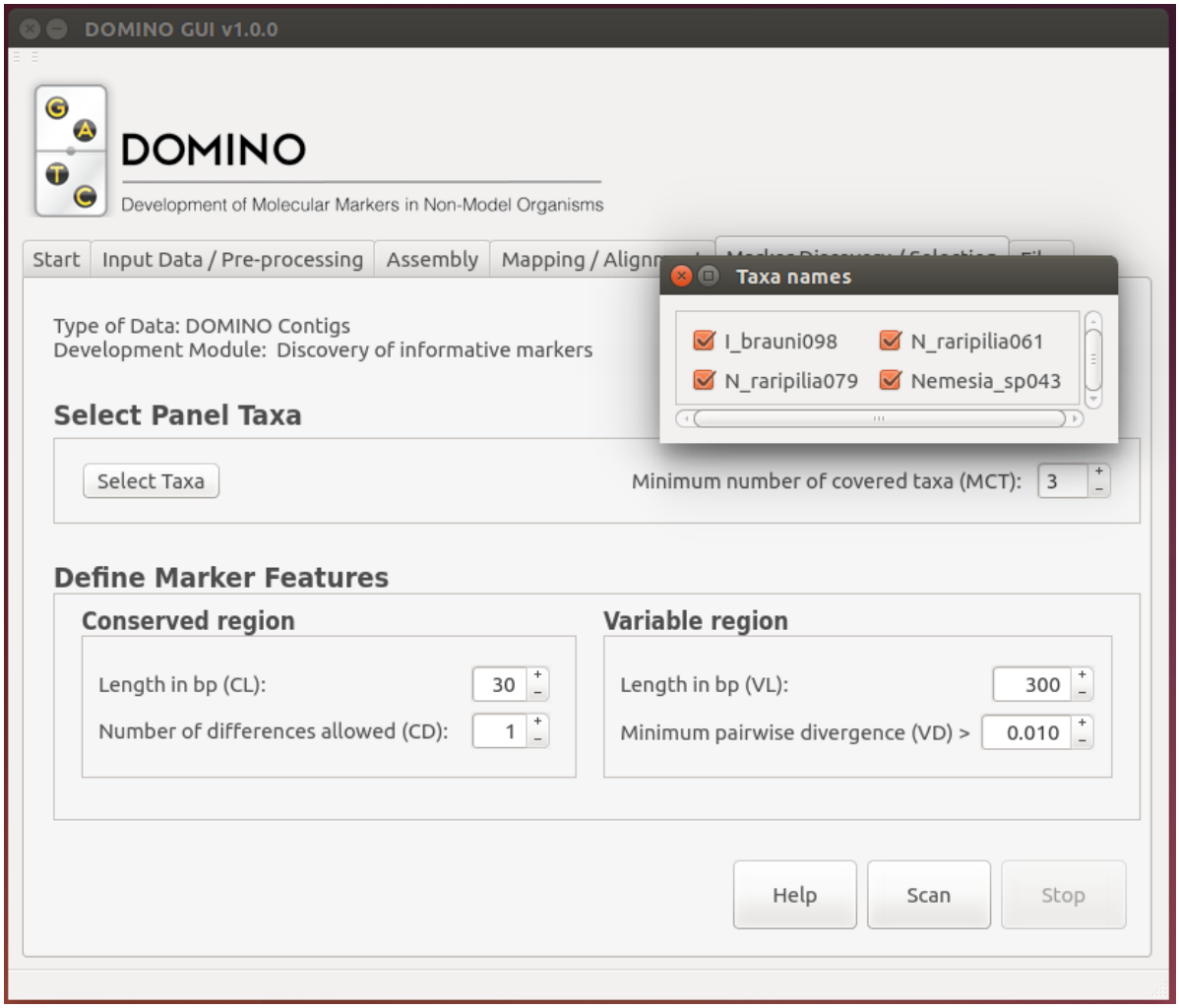
DOMINO marker discovery module
Under this module, the program will search for the presence of candidate marker regions (using a sliding window approach) across either the merged arrays of variable sites generated in current Mapping/Alignment phase or a set of pre-computed MSA loaded by the user using the MSA file(s) option in the Input Data box (Type of data).

DOMINO marker selection module
If the user chooses this module, the program uses an internal function to select the markers with the desired features among a set of pre-computed MSA. These MSA can be loaded using the MSA file(s) or the RAD-MSA files options in the Input Data box (Type of data).

Additional Information

The mapping and the building of the profiles of variable positions are performed using Bowtie2 software, SAMtools suite and new developed functions implemented in a Perl script (DM_MarkerScan.pl PERL script). Please read the documentation provided by these software and their description in the DOMINO paper (Frías-López *et al.* 2016) for details on parameters and options. See the DOMINO website to see some examples of the input file formats accepted by DOMINO.

7 DOMINO ––Marker Discovery/Selection



DOMINO searches for informative markers (i.e., regions of a particular length with a desirable level of nucleotide variation among selected taxa or encompassing a minimum number of them (which can be optionally flanked by two highly conserved regions), in the merged arrays of variable sites generated in the previous phase.

Since DOMINO can search markers based on profiles built using different reference sequences (e.g. four in our example, one for each taxon), the same region can be identified/selected as a marker more than once. To avoid reporting this redundant information, DOMINO uses BLAST to identify and collapse these redundant markers across the different profiles of variable sites and report only one of them.

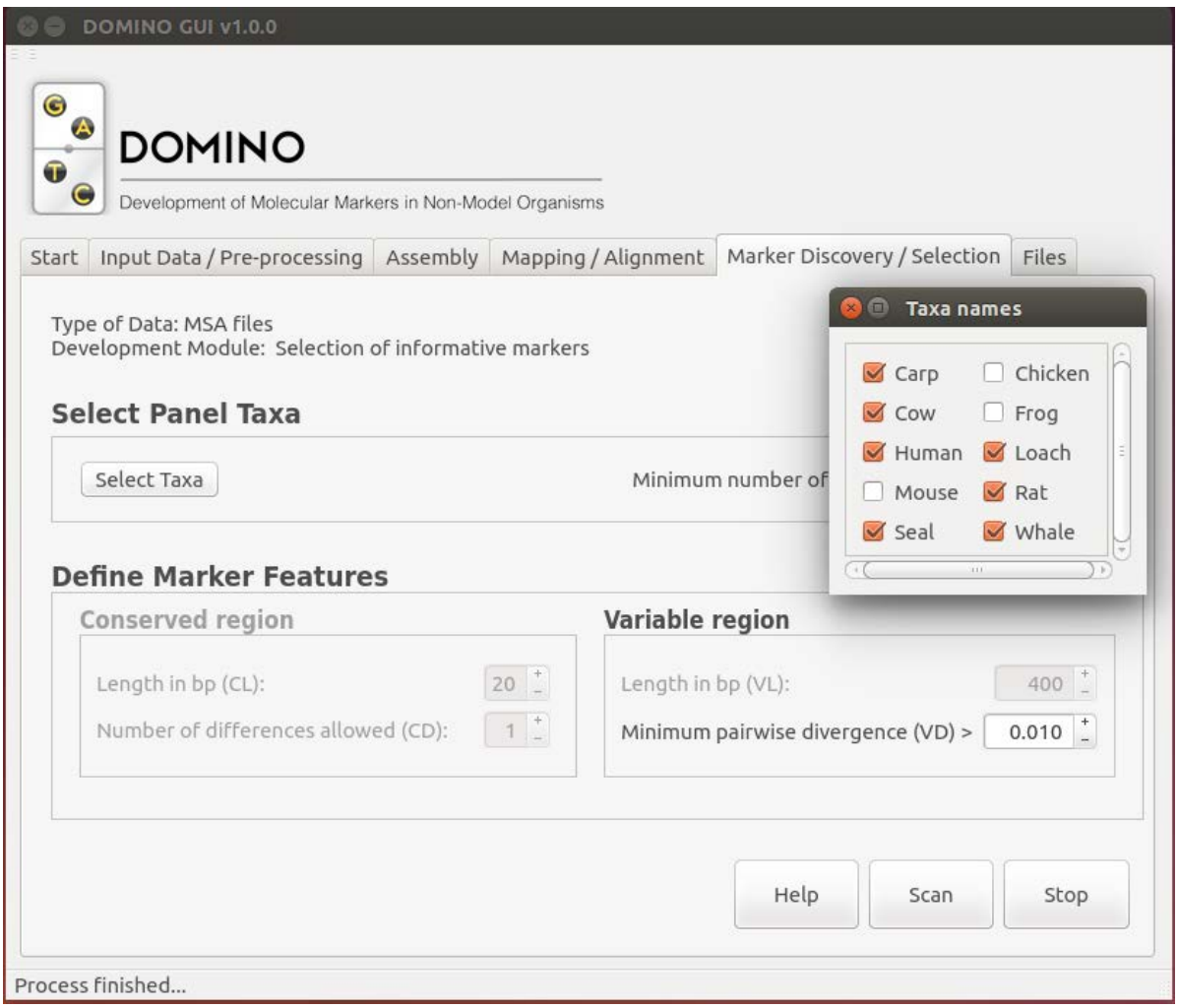
Panel Taxa box

Regardless of the number of taxa included in the input data file (taxa panel), the user can select in this box the taxa to be used in the Marker Discovery/Selection phase, and therefore to restrict the marker search to all or a subset of them (with a minimum of two). Moreover, DOMINO allows specifying the minimum number of taxa (amongst the currently selected taxa) that must satisfy the specified conditions (marker features) for a region to be considered as an informative marker (Minimum number of covered taxa, MCT). For instance, a value of MCT = 4 means that the user will restrict the marker search to aligned regions covered with information from all four taxa (as in the example). When the objective is to design markers useful for further PCR amplification and

sequencing in a larger focal taxa set (taking into account the phylogenetic relationships among the four taxa from the panel), and DOMINO detects few markers, the user might relax the search conditions by changing the MCT value. For instance, setting the MCT value to 3, DOMINO will search for markers in regions covered by at least 3 of the 4 selected taxa, and in which at least 3 of them exhibit the minimum variation desired for the marker (see the description of VD parameter for details). In case of using RAD data, DOMINO will ensure that, in each RAD loci, the number of variable positions between at least 3 taxa is within the required range (see the description of VP parameter for details). Hence, this option is very useful to find markers informative to resolve the phylogenetic relationships between or among specific groups (i.e. markers informative at different phylogenetic/population genetics ranges).

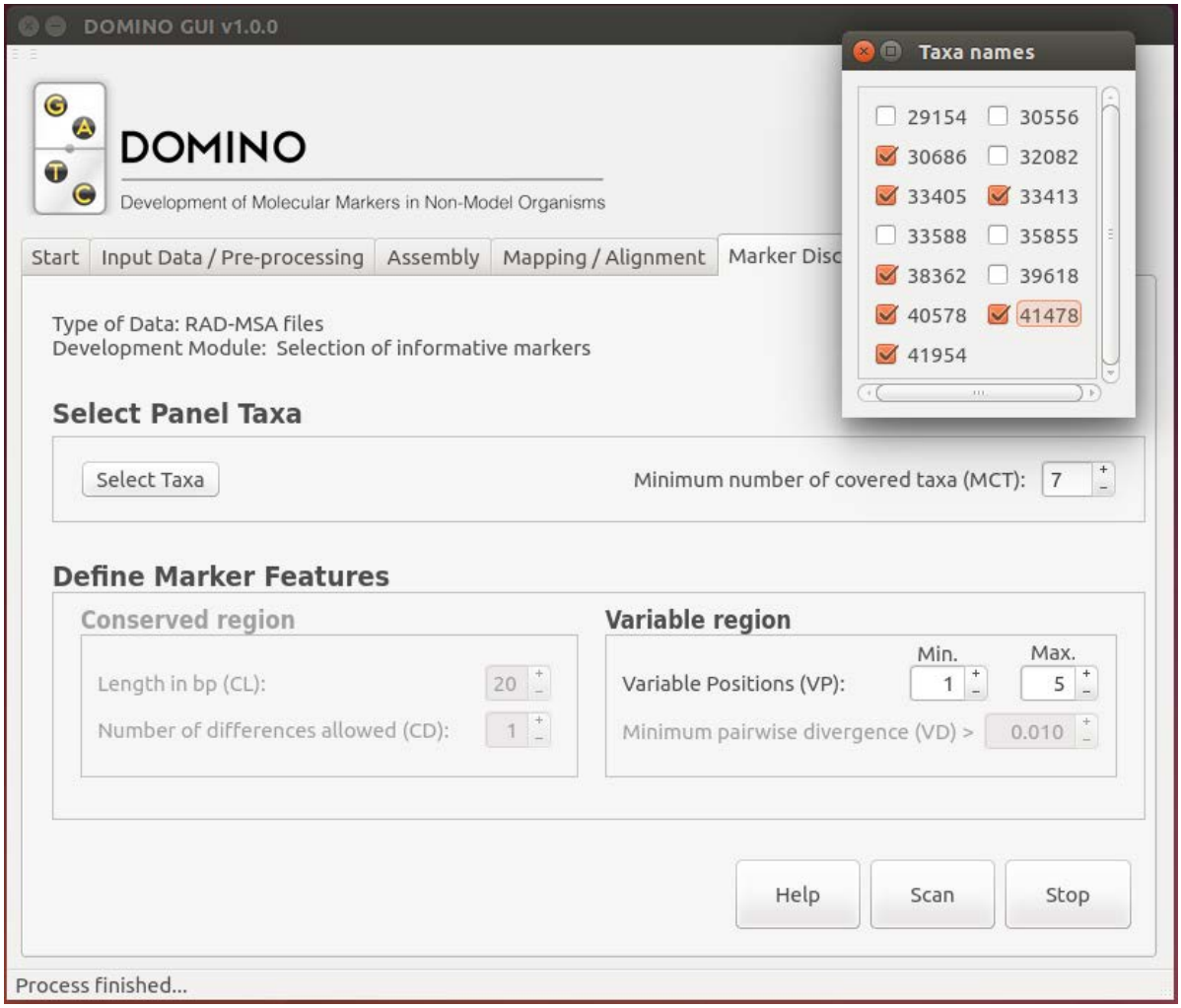
Define Marker Features

This box allows choosing a large number marker features. If the user are interested in obtaining informative markers for their further PCR amplification and sequencing in other phylogenetically related taxa (focal taxa), DOMINO will search for conserved stretches flanking the identified or selected marker; in this case the search can be restricted to conserved regions of a specified Length (CL), and exhibiting a maximum Number of nucleotide differences across taxa (CD). For the marker region itself, the user can search for a particular Length range (VL), and restrict the analyses to regions exhibiting a minimum level of variation (nucleotide substitutions per site) between any pair of taxa (VD value). Furthermore, using the Polymorphic variants option, DOMINO will use polymorphic positions to build the profile of variable sites.



RAD-MSA data and similar data

If the user enters RAD loci data (Custom Run option; see the Mapping/Alignment section), DOMINO allows selecting the most informative markers (RAD loci) in the same way and with the same definable features described above. In this case, however, a specific range of variable positions (VP) between the closest taxa instead of a minimum pairwise divergence level should be specified. The latter option will allow selecting informative RAD loci while excluding all cases exhibiting anomalous high levels of variation (which might reflect RAD tag clustering errors).



Scan / Re-Scan button

Use the scan button to start the Marker Discovery/Alignment phase. The Re-Scan button will be automatically activated after finished the first run; this button allows the user change some maker feature parameters to repeat the search analysis.

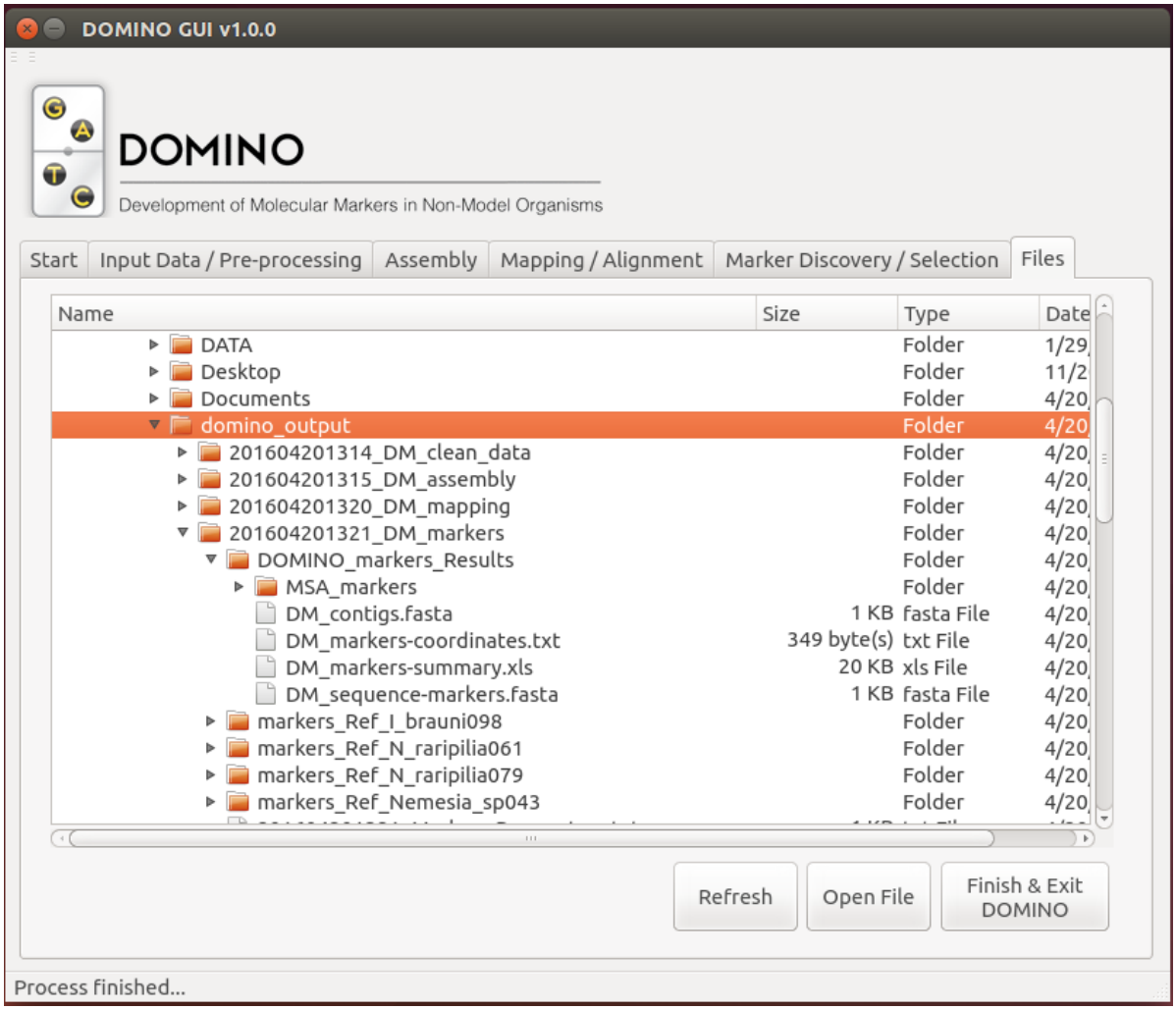
Results of our example data file

Under the Full DOMINO Run option and using the default parameter values for the previous tabs, DOMINO identifies 16 markers in 10 independent contigs (CL=30, CD=1, VL=300, VD=0.01, MCT=3); see the `4000Nemesia_Example_output.xls`, in the DOMINO website. See also the Frías-López *et al.* 2016 (supplementary Tables S4-S5) for the outcome for other parameter combinations.

Additional Information

The marker discovery and selection phase is mainly performed by our new developed functions implemented in the Perl script `DM_MarkerScan.pl`, although it also uses other scripts and pieces of software such as BLAST. Please read the paper describing DOMINO (Frías-López *et al.* 2016), for details on parameters and options. See the DOMINO website to see some examples of the input file formats accepted by DOMINO.

8 DOMINO --OUTPUT



In this TAB the user can easily visualize and get access to the folders and files that contains the results and all the stuff form intermediate analyses.

List of informative markers (`DOMINO_markers_Results` folder)

`DM_markers-summary.txt` and `DM_markers-summary.xls` contain (in plain text and Excel format) the relevant information about the markers discovered or selected by DOMINO. This information also includes, if requested, the coordinates of the conserved regions to be used for further PCR amplification and sequencing experiments.

The `DM_sequence-markers.fasta` file contains the DNA sequence of the complete region identified or selected as an informative marker in a multi-FASTA format (including the conserved regions if applicable). The `DM_contigs.fasta` file contains the DNA sequence of all contigs with one or more identified markers in a multi-FASTA.

In the sub-folder `MSA_markers`, DOMINO stores the MSA files of each identified or selected informative marker in FASTA format. In this case, the MSA will always contain only the marker region itself, regardless of whether the flanking conserved regions were requested or not. Furthermore, DOMINO also provides a file with all these MSA concatenated in FASTA format (markers are separated by a white space), which can be directly used for downstream phylogenetic or population genetic analyses.

9 Acknowledgements & Funding

This work was supported from the Ministerio de Economia y Competitividad of Spain (grants BFU2010-15484 and CGL2013-45211 to J.R., and CGL2012-36863 to M.A.A.), from the Comissió Interdepartamental de Recerca i Innovació Tecnològica of Spain (2009SGR-1287; 2014SGR1055; 2014SGR1604). J.R. and M.A.A. were partially supported by ICREA Academia (Generalitat de Catalunya), A.S-G. by a Beatriu de Pinós postdoctoral fellowship (Generalitat de Catalunya), C.F-L by an IRBio predoctoral fellowship (Universitat de Barcelona) and J.F.S-H by a FPU predoctoral fellowship (Ministerio de Educación y Ciencia).