QDD

For Windows and Linux
Version 1 (2009)

A user-friendly program to select microsatellite markers and design primers from large sequencing projects

Emese Meglécz[1] and Jean-François Martin[2]

[1]Aix-Marseille Université, CNRS, IRD, UMR 6116 – IMEP, Equipe EGE Case 36, 3 Place Victor Hugo, 13331
Marseille Cedex 3,
France

[2]Montpellier SupAgro, INRA, CIRAD, IRD, Centre de Biologie et de Gestion des Populations, Campus
International de Baillarguet, CS30016, 34988 Montferrier-sur-Lez, France

emese.meglecz@univ-provence.fr
http://www.up.univ-mrs.fr/Local/egee/dir/meglecz/QDD.html

In this manual, we tried to give many details and instructions so that end-users without experience of using programs by command lines would find their way. However, we are aware that most users have plenty of other things to do than reading a 13 page manual. So you can start by reading the highlighted text. If you stuck, then read the rest.

1. Overview

Large scale sequencing has become affordable; therefore it is likely to replace rapidly microsatellite isolation involving cloning. Apart from cost- and labour-efficiency, access to a large number of sequences has two great advantages:

(i) Microsatellite selection can be more stringent. Using only microsatellites that are not compound or interrupted, thus likely to follow a simple mutation model, provide markers that are more easily interpretable.

(ii) Microsatellite amplification by PCR can be seriously affected by microsatellite and mobile element associations. The detection of large sequence clusters can suggest the presence of mobile elements, and thus eliminating microsatellites that are found in these clusters can increase the proportion of working primers compared to the total number of primers tested.

QDD is designed to treat all bioinformatics steps from raw sequences until obtaining PCR primers: sorting sequences by tag, removing adapters/vectors, detection of microsatellites, detection of redundancy/possible mobile element association, selection of sequences with target microsatellites and sufficiently long nanosatellite-free flanking regions, and primer design.

A user-friendly windows interface i-QDD is under development. The current version can be run both under Linux and Windows in an easy to use command line option.

2. Glossary

Genomic multicopies: Loci present more than once in the genome. They can be either the results of duplication events or transposition.

Flanking region: The whole sequence apart from perfect microsatellites. This simple definition can be applied, since the lengths of the reads are compatible with PCR, thus it is not necessary to define a maximum for length of a flanking region.

Soft masking in BLAST: BLAST prevents seeding (starting the alignment by a perfect match of a predefined length) in masked regions, but allows alignment extension through them if soft masking is applied.

Tag: A short DNA stretch added at the 5'-end of the DNA fragment to be sequenced for identification. Different tags can be added to DNA from different sources (e.g. species) and the pooled DNA is loaded on a non-fractioned PicoTiter plate, thus gaining space and quantities of reads. Sequences coming from different sources are identified according to their tag.

Perfect microsatellite: Microsatellite composed of one single motif with no interruption. The minimum number of repetition is defined by the user.

Nanosatellite: Tandem repetition of 1-6 bp motifs, where the number of repetition is lower than the number of repetitions of microsatellites (Very short microsatellites).

3. Installation

QDD is written in Perl and is run as a standalone application on Windows or Linux systems.

For both versions the following freely available programs should be installed in order to be able to run QDD:

ActivePerl (http://www.activestate.com/activeperl/)

BLAST (ftp://ftp.ncbi.nih.gov/blast/executables/)

ClustalW (ftp://ftp.ebi.ac.uk/pub/software/clustalw2/) Use clustalw2 and not formerly widely used clustalw1.83.

Primer3 (http://primer3.sourceforge.net/)

3.1. Install ActivePerl, BLAST, ClustalW2 and Primer3 (The path to these executables should not contain a space)

Important: If you are working on MS Windows install Clustal2 using the msi file and keep the files within the folder selected during the installation process

If you are working on Linux install the package csv_xs (sudo apt-get install libtext-csv-xs-perl)

3.2. Untar and unzip QDD.tar.gz for Linux, extract QDD.rar (by WinRar) for windows

Put all files into one folder

3.3. Make a project folder for the input files

4. Description

QDD is composed of three parts. Each of them can be run separately.

4.1 Sequence cleaning and microsatellite detection: pipe1.pl

Most of the steps do not take longer than a few minutes. If there are a million of sequences in the tag sorting step, it can take about 30 minutes.

4.1.1.Input files

All input files must be in the project folder that does not contain other fasta files. The name of the input folder is set by the user in the parameters (see 4.2).

From here onwards we give the names of the output files for a run where the original input fasta file was named ‘sample.fas’ and put into a folder project ‘data’ that is a subfolder within QDD. (data\sample.fas)

4.1.1.1. tag.fas (must be named 'tag.fas'; fasta file with all tag sequences; optional)

e.g.

```
>MID1
ACGAGTGCGT
>MID2
ACGCTCGACA
```

4.1.1.2. adapter.fas (must be named 'adapter.fas'; fasta file with all adapters/vectors that might be present in the sequences; optional but STRONGLY recommended where adapters apply)

4.1.1.3. fasta files from the sequencing project

There might be more than one file. The program deals with them one after the other.

The name of the fasta files can have any alphanumerical characters and underscore and must have ‘.fas’ extention (*e.g.* sample.fas). Everything in the definition line after ‘>’ and before the first space is read as the sequence identifier. The identifier can have any alphanumerical characters and underscore. Replace all other characters by underscore.

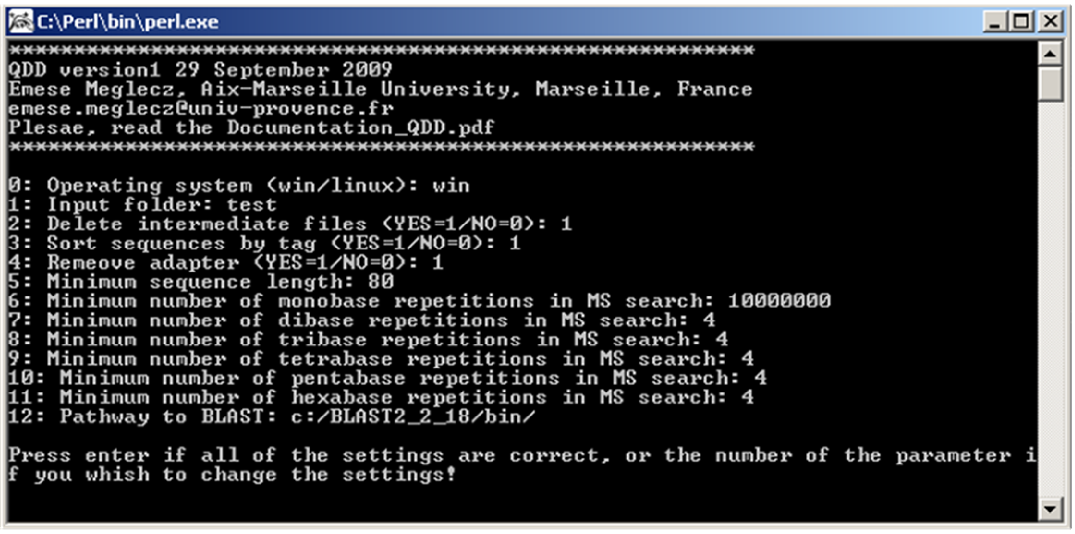
e.g.

```
>FVU26NR06DGVOE
ACGAGTGCGTGCCTAGCTAGCAGAATCACACACACACACACACACACACACTATGTACT
CTCCTTTGTGAAATACATACGACATGTGTACGTAAACAACACT
>FVU26NR06DIOGK
ACGAGTGCGTAAGGCCTAGCTAGCAGAATCGTTTCCTAATGATGCGCTTCCAAAACTACTCT
CTGTGCGACTCTTTAACCTT
…
```

4.1.2. Steps of pipe1.pl

4.1.2.1. It identifies and removes tags and writes one fasta file per tag with the tag free sequences (plus 1 file with sequences that did not have detectable tag). Optional.
4.1.2.2. Removes adaptors/vectors Optional. If adapter is not found at the beginning of the sequence, the sequence is removed.
4.1.2.3. Selects sequences longer or equal than the user-defined limit
4.1.2.4. Selects sequences that contain microsatellites.
4.1.2.5. The main output files are called data\data2\sample_tag1_pipe2.fas, data\data2\sample_tag2_pipe2.fas, data\data2\sample_NOTAG_pipe2.fas (one file per tag, where tagX is the name of the different tags) and contains all sequences that have microsatellites and are longer than a user defined limit. It is placed into ‘data2’ subfolder of your project folder and will be the input for pipe2.pl. All intermediate files are printed into subfolders ‘data\sample\’ (a single subfolder per input file).

4.1.3. Parameters of pipe1.pl with default values



```
**********************************************************
QDD version1 29 September 2009
Emese Meglecz, Aix-Marseille University, Marseille, France
emese.meglecz@univ-provence.fr
Plesae, read the Documentation_QDD.pdf
**********************************************************

0: Operating system (win/linux): win
1: Input folder: test
2: Delete intermediate files (YES=1/NO=0): 1
3: Sort sequences by tag (YES=1/NO=0): 1
4: Remeove adapter (YES=1/NO=0): 1
5: Minimum sequence length: 80
6: Minimum number of monobase repetitions in MS search: 10000000
7: Minimum number of dibase repetitions in MS search: 4
8: Minimum number of tribase repetitions in MS search: 4
9: Minimum number of tetrabase repetitions in MS search: 4
10: Minimum number of pentabase repetitions in MS search: 4
11: Minimum number of hexabase repetitions in MS search: 4
12: Pathway to BLAST: c:/BLAST2_2_18/bin/

Press enter if all of the settings are correct, or the number of the parameter i
f you whish to change the settings!
```

4.1.3.1. Operating system (win/linux): win
4.1.3.2. Input folder: e.g. data
If the input folder is not the subfolder of the folder that contains the QDD scripts, the whole path should be specified. (e.g. c:\data). Only alphanumerical characters and underscore is allowed in the name
4.1.3.3. Delete intermediate files (YES=1/NO=0): default = 1
If 1 keeps only important intermediate files, if it is set to 0 all intermediate files are kept (option used for fine intermediate statistics or troubleshooting, otherwise delete intermediate files is preferred)
4.1.3.4. Sort sequences by tag (YES=1/NO=0): default = 1
If 1, QDD scans for tags defined in tag.fas, otherwise skips the tag sorting step
4.1.3.5. Remove adapter (YES=1/NO=0): default = 1
If 1, QDD removes vector/adapter sequences. Attention! If adapters/vector/tags are not removed when they should, many sequences are unnecessarily eliminated by pipe2.pl (see 4.2.). Therefore skip these steps only if you are sure that you have a clean dataset.
4.1.3.6. Minimum sequence length: default = 80
Keeps sequences longer than 80 bp (without adapter and tag)
4.1.3.7. Minimum number of monobase repetitions in MS search: default = 1000000
4.1.3.8. Minimum number of dibase repetitions in MS search: default = 4
4.1.3.9. Minimum number of tribase repetitions in MS search: default = 4
4.1.3.10. Minimum number of tetrabase repetitions in MS search: default = 4
4.1.3.11. Minimum number of pentabase repetitions in MS search: default = 4
4.1.3.12. Minimum number of hexabase repetitions in MS search: default = 4
In 4.1.3.7-12 Minimum number of repeats for each motif length during MS search, only sequences that have at least one perfect microsatellite of at least X repetitions are kept. It is better to keep the number of repetitions low at this step, to keep as many sequences as possible for the following analyses. More stringent selection on the length of the target microsatellites will be possible at Stage3.
4.1.3.13. Pathway to BLAST: pathway to BLAST executables. Attention, executables are usually

found in the \bin\ subfolder of BLAST, therefore it must be included on the path (e.g. c:\BLAST2_2_18\bin\)

4.2. Sequence similarity detection: pipe2.pl

This stage eliminates redundancy in the widest sense: copies of the same locus, sequences that potentially have more than one copies in the genome. This stage is the most time-consuming. The time of the run can vary from a few minutes to a few hours, and it depends on the number of sequences and the degree of redundancy (including intra genomic repetitions) of the data.

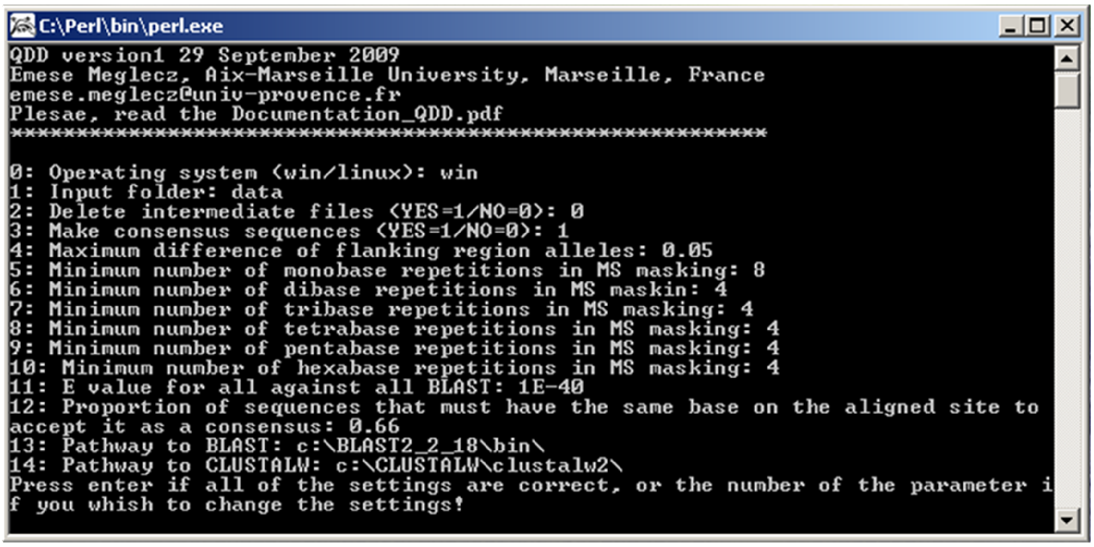
4.2.1. Input files

This stage can treat input files with up to 50 000 sequences in a single fasta file.
The input files were prepared by pipe1.pl and found in the data2 subfolder of the original input folder 'data'

4.2.2. Steps of pipe2.pl

4.2.2.1. Detects sequence similarity by an all-against-all BLAST
4.2.2.2. Eliminates sequences that have more than 1 blast hit between the two same sequences (multihit, possibly minisatellites)
4.2.2.3. Removes/concatenates 100% identical sequences
4.2.2.4. Calculates pair wise identity along the whole flanking regions if similarity was detected by BLAST
4.2.2.5. Establishes contigs if pair wise similarity along the flanking region is higher than user-defined limit
4.2.2.6. Makes majority rule consensus sequences (consensus coefficient is user defined)
4.2.2.7. All against all BLAST of file containing all consensuses sequences plus sequences that have blast hits but not included in the contig
4.2.2.8. Selects consensuses sequences that did not have hit to any other sequence in the previous BLAST
4.2.2.9. Prepares a file with selected consensuses sequences and all original 'unique' sequences (either did not have a BLAST hit or only to sequences 100% identical. This is placed into 'data3' subfolder and will be the input file of pipe3.pl
4.2.2.10. All intermediate files are placed into a 'data\sample_tag1_pipe2\' subfolder (one folder per input file). The final fasta files with sequences selected for primer design are placed into the 'data3' subfolder of your project folder (e.g. data\data3\sample_tag1_pipe3.fas). These files will be the input files for pipe3.pl.

4.2.3. Parameters of pipe2.pl with default values



```
QDD version1 29 September 2009
Emese Meglecz, Aix-Marseille University, Marseille, France
emese.meglecz@univ-provence.fr
Plesae, read the Documentation_QDD.pdf
**********************************************************

0: Operating system (win/linux): win
1: Input folder: data
2: Delete intermediate files (YES=1/NO=0): 0
3: Make consensus sequences (YES=1/NO=0): 1
4: Maximum difference of flanking region alleles: 0.05
5: Minimum number of monobase repetitions in MS masking: 8
6: Minimum number of dibase repetitions in MS maskin: 4
7: Minimum number of tribase repetitions in MS masking: 4
8: Minimum number of tetrabase repetitions in MS masking: 4
9: Minimum number of pentabase repetitions in MS masking: 4
10: Minimum number of hexabase repetitions in MS masking: 4
11: E value for all against all BLAST: 1E-40
12: Proportion of sequences that must have the same base on the aligned site to
accept it as a consensus: 0.66
13: Pathway to BLAST: c:\BLAST2_2_18\bin\
14: Pathway to CLUSTALW: c:\CLUSTALW\clustalw2\
Press enter if all of the settings are correct, or the number of the parameter i
f you whish to change the settings!
```

4.2.3.1. Operating system (win/linux): win
4.2.3.2. Input folder: data
Must be the same as for pipe1.pl. This folder contains the subfolder data2, with the input files of pipe2.pl
4.2.3.3. Delete intermediate files (YES=1/NO=0): default = 1
4.2.3.4. Make contigs and consensus sequences (if 1), else uses only sequences that had no Blast hit, or

only to redundant sequences. Unless you have a very large input file (more than 50 000 sequences) it is better to use option 1. If the run time is excessively long you can consider running option 0 (no contigs are prepared), and use only unique sequences in subsequent analyses.

4.2.3.5. Maximum difference of flanking region alleles: default = 0.05

While making contigs a sequence is added to a contig if its flanking region similarity to at least one of the existing sequence in the contig is greater than 0.95

4.2.3.6. Minimum number of monobase repetitions in MS masking: default = 8

4.2.3.7. Minimum number of dibase repetitions in MS masking: default = 4

4.2.3.8. Minimum number of tribase repetitions in MS masking: default = 4

4.2.3.9. Minimum number of tetrabase repetitions in MS masking: default = 4

4.2.3.10. Minimum number of pentabase repetitions in MS masking: default = 4

4.2.3.11. Minimum number of hexabase repetitions in MS masking: default = 4

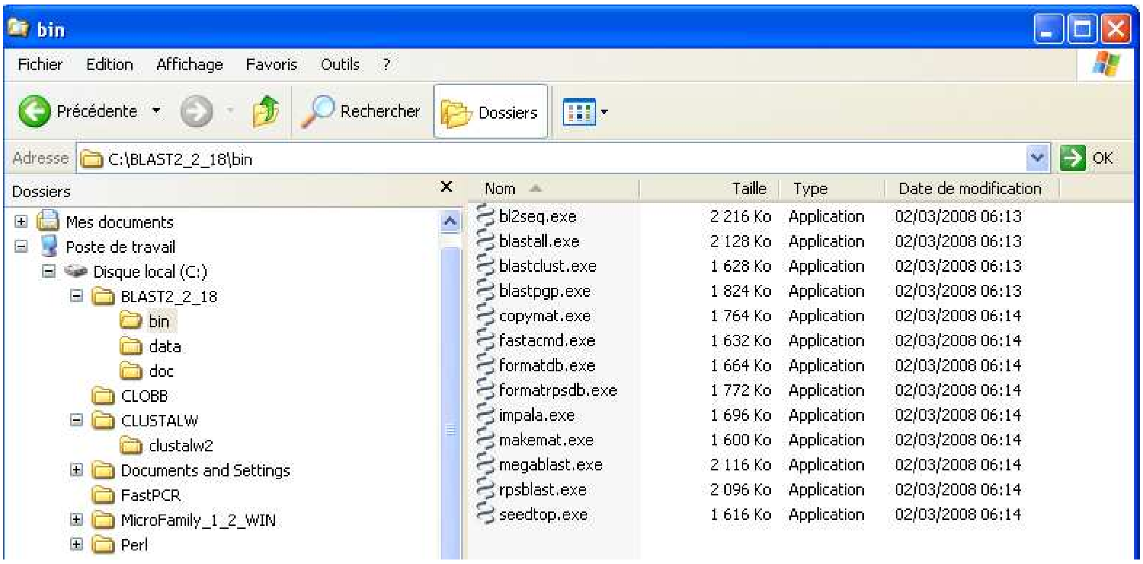
In 4.2.3.5-10: Before the BLAST the microsatellites are soft masked, if they have at least the given number of repetitions

4.2.3.12. E value for all against all BLAST: default = 1E-40

4.2.3.13. Proportion of sequences that must have the same base on the aligned site to accept it as a consensus: default = 0.66. Sequences of the contigs are aligned. For each site, a nucleotide is accepted as a consensus if it is present in more than 66% of the informative bases (not N) on that site. Otherwise N is put in the consensus sequence of the given site.

4.2.3.14. Pathway to BLAST: e.g. c:\BLAST2_2_18\bin\

4.2.3.15. Pathway to CLUSTALW: e.g. c:\CLUSTALW\clustalw2\



Adresse C:\BLAST2_2_18\bin

Nom	Taille	Type	Date de modification
bl2seq.exe	2 216 Ko	Application	02/03/2008 06:13
blastall.exe	2 128 Ko	Application	02/03/2008 06:13
blastclust.exe	1 628 Ko	Application	02/03/2008 06:13
blastpgp.exe	1 824 Ko	Application	02/03/2008 06:13
copymat.exe	1 764 Ko	Application	02/03/2008 06:14
fastacmd.exe	1 632 Ko	Application	02/03/2008 06:14
formatdb.exe	1 664 Ko	Application	02/03/2008 06:14
formatrpsdb.exe	1 772 Ko	Application	02/03/2008 06:14
impala.exe	1 696 Ko	Application	02/03/2008 06:14
makemat.exe	1 600 Ko	Application	02/03/2008 06:14
megablast.exe	2 116 Ko	Application	02/03/2008 06:14
rpsblast.exe	2 096 Ko	Application	02/03/2008 06:14
seedtop.exe	1 616 Ko	Application	02/03/2008 06:14

4.3. Microsatellite selection and primer design: pipe3.pl selects sequences with microsatellites longer than a user defined limit, and that have a sufficiently long nanosatellite-free flanking region. Runs Primer3 and rewrites its outfile in an easy to use format (tbl). This stage takes a few minutes only.

4.3.1 Input files

The input files were prepared by pipe2.pl and found in the data3 subfolder of the original input folder. (e.g. data\data3\sample_tag1_pipe3.fas)

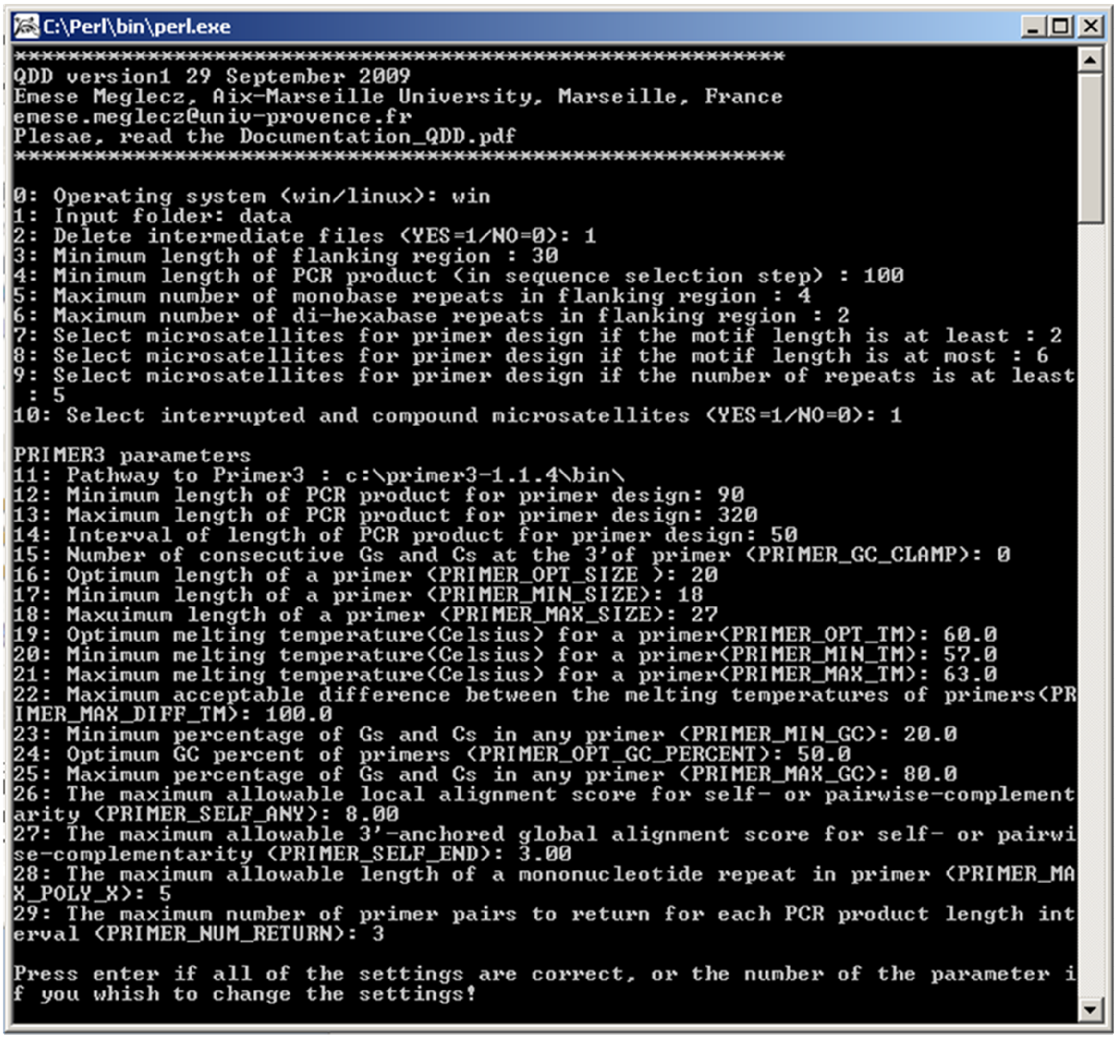
4.3.2 Steps of pipe3.pl

4.3.2.1. Selects sequences that have at least a given number of repeats, a nanosatellite-free flanking region of a given length, and a possibility to have a nanosatellite-free PCR product of a given length.

4.3.2.2. Prepares an input file for Primer3 and a fasta file with all target and nanosatellite printed in lower case. First and last positions and motifs of each micro- and nanosatellites are printed in the definition line

4.3.2.3. Runs Primer3 for each user defined PCR product length interval. Most of the parameters for Primer3 can be set directly by a menu (see 4.3.3. for details) The target region (target microsatellite) and excluded region (till the last nanosatellite before the target microsatellite and from the first nanosatellite after the target microsatellite) are defined automatically by QDD. All primer pairs and their descriptions are printed in a table, as well as the motif, length and position of the target microsatellite.

4.3.3. Parameters of pipe3.pl with default values



```
**********************************************************
QDD version1 29 September 2009
Emese Meglecz, Aix-Marseille University, Marseille, France
emese.meglecz@univ-provence.fr
Plesae, read the Documentation_QDD.pdf
**********************************************************

0: Operating system (win/linux): win
1: Input folder: data
2: Delete intermediate files (YES=1/NO=0): 1
3: Minimum length of flanking region : 30
4: Minimum length of PCR product (in sequence selection step) : 100
5: Maximum number of monobase repeats in flanking region : 4
6: Maximum number of di-hexabase repeats in flanking region : 2
7: Select microsatellites for primer design if the motif length is at least : 2
8: Select microsatellites for primer design if the motif length is at most : 6
9: Select microsatellites for primer design if the number of repeats is at least
 : 5
10: Select interrupted and compound microsatellites (YES=1/NO=0): 1

PRIMER3 parameters
11: Pathway to Primer3 : c:\primer3-1.1.4\bin\
12: Minimum length of PCR product for primer design: 90
13: Maximum length of PCR product for primer design: 320
14: Interval of length of PCR product for primer design: 50
15: Number of consecutive Gs and Cs at the 3'of primer (PRIMER_GC_CLAMP): 0
16: Optimum length of a primer (PRIMER_OPT_SIZE ): 20
17: Minimum length of a primer (PRIMER_MIN_SIZE): 18
18: Maxuimum length of a primer (PRIMER_MAX_SIZE): 27
19: Optimum melting temperature(Celsius) for a primer(PRIMER_OPT_TM): 60.0
20: Minimum melting temperature(Celsius) for a primer(PRIMER_MIN_TM): 57.0
21: Maximum melting temperature(Celsius) for a primer(PRIMER_MAX_TM): 63.0
22: Maximum acceptable difference between the melting temperatures of primers(PR
IMER_MAX_DIFF_TM): 100.0
23: Minimum percentage of Gs and Cs in any primer (PRIMER_MIN_GC): 20.0
24: Optimum GC percent of primers (PRIMER_OPT_GC_PERCENT): 50.0
25: Maximum percentage of Gs and Cs in any primer (PRIMER_MAX_GC): 80.0
26: The maximum allowable local alignment score for self- or pairwise-complement
arity (PRIMER_SELF_ANY): 8.00
27: The maximum allowable 3'-anchored global alignment score for self- or pairwi
se-complementarity (PRIMER_SELF_END): 3.00
28: The maximum allowable length of a mononucleotide repeat in primer (PRIMER_MA
X_POLY_X): 5
29: The maximum number of primer pairs to return for each PCR product length int
erval (PRIMER_NUM_RETURN): 3

Press enter if all of the settings are correct, or the number of the parameter i
f you whish to change the settings!
```

4.3.3.1. Operating system (win/linux): win
4.3.3.2. Input folder: e.g. data
Must be the same as for pipe1.pl. This folder contains the subfolder data3, with the input files of pipe3.pl
4.3.3.3. Deletes intermediate files (YES=1/NO=0): default = 1
4.3.3.4. Minimum length of flanking region: default = 30
During the sequence selection step, sequences that have at least 30 bp of nanosatellite-free flanking region on both sides of the target microsatellite are selected
4.3.3.5. Minimum length of PCR product (in sequence selection step): default = 100
During the sequence selection step, sequences that have at least 100 bp of nanosatellite-free regions (including the target microsatellite) are selected
4.3.3.6. Maximum number of monobase repeats in flanking region: default = 4
Must be greater than 3 and cannot be greater than the minimum number of repeat for target microsatellites (*4.3.3.10)*. During the sequence selection step, maximum 4 repetitions of a monobase motif is allowed in the flanking region
4.3.3.7. Maximum number of di-hexabase repeats in flanking region: default = 2;
Cannot be greater than the minimum number of repeat for target microsatellites (*4.3.3.10)*. During the sequence selection step, maximum 2 repetition of a di-hexabase motifs are allowed in the flanking region
The default parameters of *4.3.3.6.*and *4.3.3.7.*are very stringent. It reduces strongly the number of target microsatellites, but the markers obtained this way are more likely to produce a clear pattern with one repeat motif length between consecutive alleles., By setting *4.3.3.6.*and *4.3.3.7* higher you can easily obtain more markers, for the cost of having a more complicated mutation pattern.
4.3.3.8. Select microsatellites for primer design if the motif length is at least: default = 2
The motif length of the target microsatellite must be at least 2

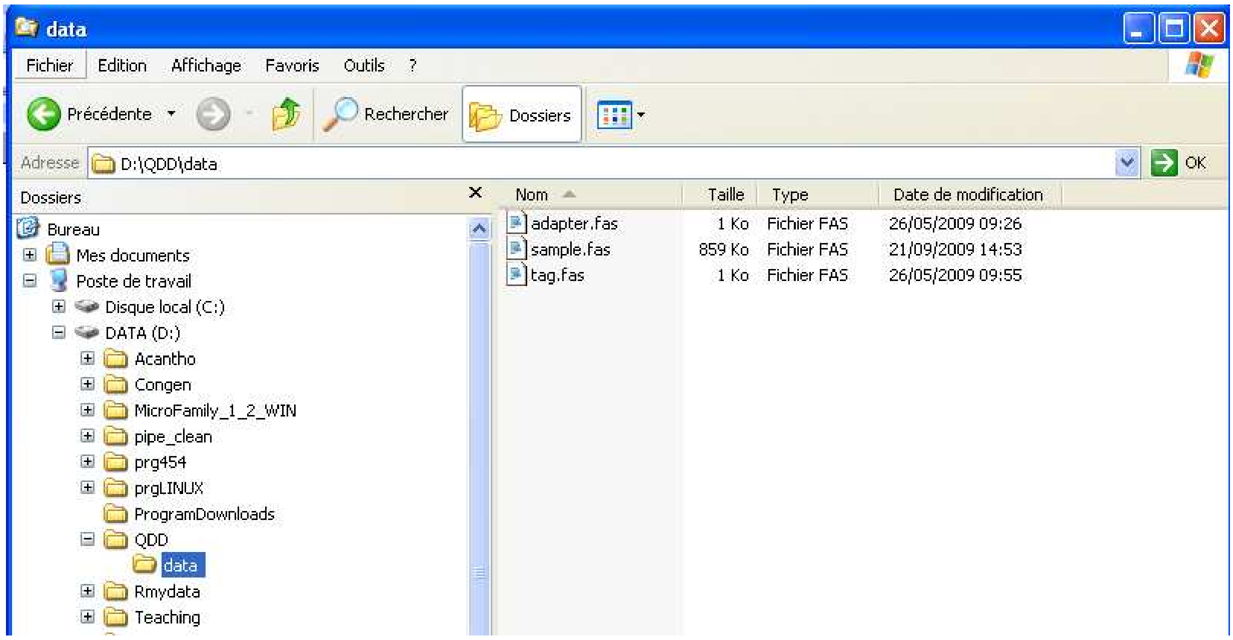
4.3.3.9. Select microsatellites for primer design if the motif length is at most: default = 6
The motif length of the target microsatellite must be at most 6; (This parameter cannot be set higher than 6)
4.3.3.10. Select microsatellites for primer design if the number of repeats is at least: default = 5
Target microsatellite must have at least 5 repeats
4.3.3.11. Select interrupted and compound microsatellites (0=NO/1=YES)
If this option is selected, interrupted and compound microsatellites (max. 2 bp between two perfect microsatellite) can also be picked as target microsatellites. For interrupted and compound microsatellites the longest perfect microsatellite repeat number is set as length. Flanking regions are checked for nanosatellites as for perfect target microsatellites.
4.3.3.12. Minimum length of PCR product for primer design (for Primer3): default = 90
4.3.3.13. Maximum length of PCR product for primer design (for Primer3): default = 320
4.3.3.14. Interval of length of PCR product for primer design (for Primer3): default = 50
Steps 4.3.3.12-14: Primer3 is run several times. Each time the desired PCR product size is set to a different interval to cover. As a default 90-140, 140-190, 190-240, 240-290, 290-320
4.3.3.15. Pathway to Primer3: e.g. c:\primer3-1.1.4\bin\
Path to Primer3 executables from the root; attention executables are found in the ‘bin’ subfolder of a folder that contains primer3
PRIMER3 internal parameters (for detailed explanation see Primer3 documentation):
4.3.3.16 Number of consecutive Gs and Cs at the 3'of primer (PRIMER_GC_CLAMP): default =0
4.3.3.17. Optimum length of a primer (PRIMER_OPT_SIZE): default =20
4.3.3.18. Minimum length of a primer (PRIMER_MIN_SIZE): default =18
4.3.3.19. Maximum length of a primer (PRIMER_MAX_SIZE): default =27
4.3.3.20. Optimum melting temperature (Celsius) for a primer (PRIMER_OPT_TM): default =60.0
4.3.3.21. Minimum melting temperature (Celsius) for a primer (PRIMER_MIN_TM): default =57.0
4.3.3.22. Maximum melting temperature (Celsius) for a primer (PRIMER_MAX_TM): default =63.0
4.3.3.23. Maximum acceptable difference between the melting temperatures of primers (PRIMER_MAX_DIFF_TM): default =1000.0
4.3.3.24. Minimum percentage of Gs and Cs in any primer (PRIMER_MIN_GC): default =20.0
4.3.3.25. Optimum GC percent of primers (PRIMER_OPT_GC_PERCENT): default =50.0
4.3.3.26. Maximum percentage of Gs and Cs in any primer (PRIMER_MAX_GC): default =80.0
4.3.3.27. The maximum allowable local alignment score for self- or pairwise-complementarity (PRIMER_SELF_ANY): default =8.00
4.3.3.28. The maximum allowable 3'-anchored global alignment score for self- or pairwise-complementarity (PRIMER_SELF_END): default =3.00
4.3.3.29. The maximum allowable length of a mononucleotide repeat in primer (PRIMER_MAX_POLY_X): default =5
4.3.3.30. The maximum number of primer pairs to return for each PCR product length interval (PRIMER_NUM_RETURN): default =3

5. Running QDD

5.1. Linux and Windows command line

5.1.1. Put the input files of pipe1.pl into your project folder. The name of project folder can contain any alphanumerical character or underscore). All fasta files with the ‘.fas’ extension are analysed, so make sure that the project folder contains only the files you want to analyse.
Input files:
5.1.1.1. tag.fas (must have this name; fasta file with all tag sequences; optional).
5.1.1.2. adapter.fas (must have this name; fasta file with all adapters/vectors that might be present in the sequences; optional but strongly recommended).

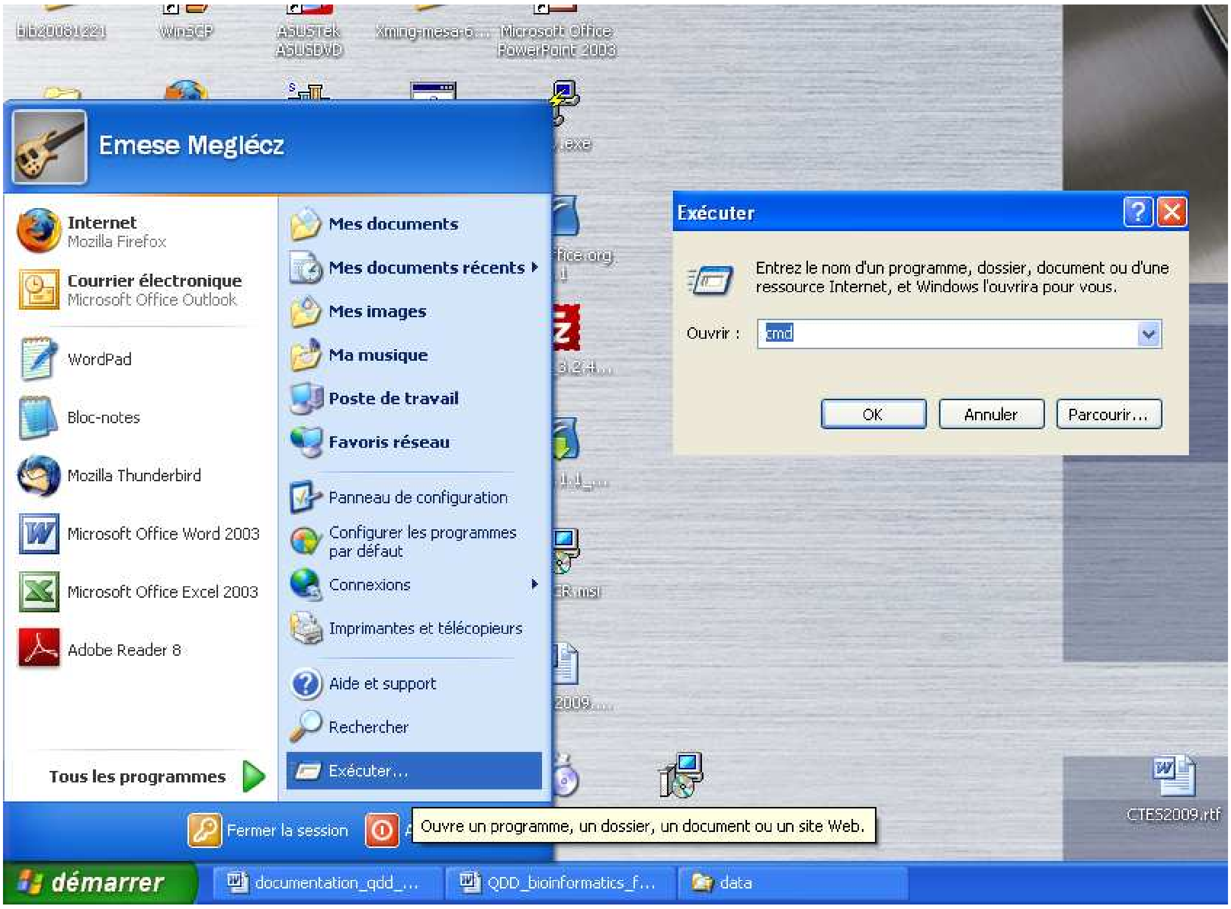
5.1.1.3. fasta files from the sequencing project. There might be more than one file. The program deals with them one after the other. The name of the fasta files can have any alphanumerical characters and underscore and must have ‘.fas’ extention (*e.g.* sample.fas). Everything in the definition line after ‘>’ and before the first space is read as the sequence identifier. The identifier can have any alphanumerical characters and underscore. Replace all other characters by ‘_’.



Adresse D:\QDD\data

Nom	Taille	Type	Date de modification
adapter.fas	1 Ko	Fichier FAS	26/05/2009 09:26
sample.fas	859 Ko	Fichier FAS	21/09/2009 14:53
tag.fas	1 Ko	Fichier FAS	26/05/2009 09:55

5.1.2. Open a terminal (START/run/cmd/OK)



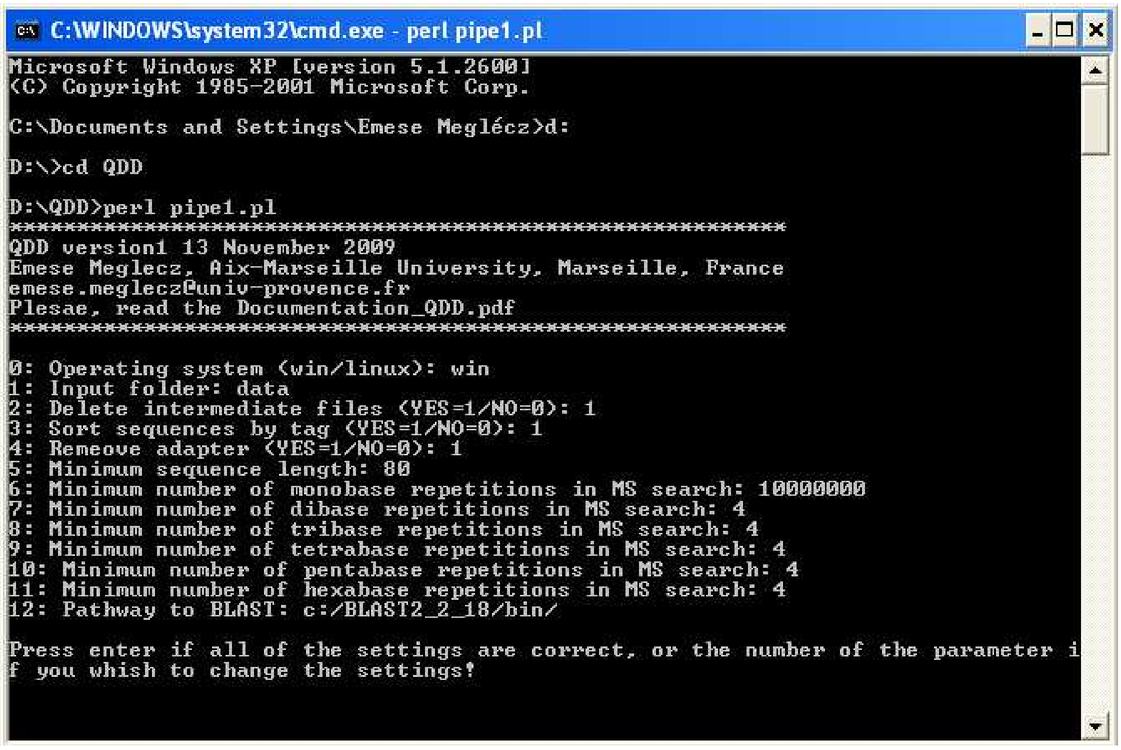
Entrez le nom d'un programme, dossier, document ou d'une ressource Internet, et Windows l'ouvrira pour vous.

Ouvrir : cmd

Ouvre un programme, un dossier, un document ou un site Web.

5.1.3. Change directory in a terminal to the folder that contains the scripts (e.g. cd d:\QDD)

5.1.4. Type ‘perl pipe1.pl’



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Emese Meglécz>d:

D:\>cd QDD

D:\QDD>perl pipe1.pl
**********************************************************
QDD version1 13 November 2009
Emese Meglecz, Aix-Marseille University, Marseille, France
emese.meglecz@univ-provence.fr
Plesae, read the Documentation_QDD.pdf
**********************************************************

0: Operating system (win/linux): win
1: Input folder: data
2: Delete intermediate files (YES=1/NO=0): 1
3: Sort sequences by tag (YES=1/NO=0): 1
4: Remeove adapter (YES=1/NO=0): 1
5: Minimum sequence length: 80
6: Minimum number of monobase repetitions in MS search: 10000000
7: Minimum number of dibase repetitions in MS search: 4
8: Minimum number of tribase repetitions in MS search: 4
9: Minimum number of tetrabase repetitions in MS search: 4
10: Minimum number of pentabase repetitions in MS search: 4
11: Minimum number of hexabase repetitions in MS search: 4
12: Pathway to BLAST: c:/BLAST2_2_18/bin/

Press enter if all of the settings are correct, or the number of the parameter i
f you whish to change the settings!
```

5.1.5. Follow the instructions on the screen to set the parameters and start the program
5.1.6. Once pipe1.pl is finished the input files for pipe2.pl are found in 'data2' subfolder of your input project folder.
5.1.7. If necessary pool some of the sample_tagX_pipe2.pl files (*e.g.* if sequences of the same species had more than 1 tag).
5.1.8. Make sure that 'data2' contains only files that you whish to analyze by pipe2.pl
5.1.9. Run pipe2.pl by typing 'perl pipe2.pl'
5.1.10. Follow the instructions on the screen to set the parameters and start the program
5.1.11. Once pipe2.pl is finished the input files for pipe3.pl are found in 'data3' subfolder of your input folder.
5.1.12. Run pipe3.pl by typing 'perl pipe3.pl'
5.1.13. Follow the instructions on the screen to set the parameters and start the program
5.1.14. The output files are found in 'sample_tagX_pipe3' subfolder of your input folder
5.1.15. If you would like to have more markers, rerun pipe3.pl with less stringent conditions. Your original tagX_pipe3 folder is automatically renamed as tagX_pipe3_1 to avoid overwriting, but it is wise to rename the folder with more explicit names, before re-running the program.

6. Description of the outfiles

Apart from the most important result files (**sample_tagX_pipe3_abc_primers.tbl** see 6.3. for definitions of abc) QDD produces a great number of outfiles. As a default value most of these are deleted at the end of the run, some of them are kept however, since it might be interesting for some users.
If the 'delete intermediate files' are set to 0, more intermediate files are kept. These are probably only of interest to the authors of QDD for troubleshooting.
All of the files are either fasta files or simple text files that can be opened by excel. The separators for the columns are either spaces (files produced by BLAST) or ';'.

At the end of the run, there are several files and subfolders in you project folder.
Your input files are still found directly in the project folder (tag.fas, adapter.fas, and fasta files with the reads e.g. sample.fas)
The data2 and data3 subfolders contain the input files for pipe2.pl and pipe3.pl respectively (produced by pipe1.pl and pipe2.pl, respectively). There might be more than one fasta files in each of them, according to the number of input files and the number of tags used. When running pipe2.pl or pipe3.pl all files in data2 or data3 are analysed one after the other.
The (supplementary) outfiles of pipe1.pl (all outfiles apart from the input files for pipe2.pl) are found in

sample subfolder (one subfolder per input read file for pipe1.pl)
The (supplementary) outfiles of pipe2.pl (all outfiles apart from the input files for pipe3.pl) are found in sample_TAGY_pipe2 subfolder (as many subfolders as files in data2)
All outfiles of pipe3.pl are found in sample_TAGY_pipe3 subfolder (as many subfolders as files in data3)

6.1. Pipe1.pl

6.1.1 sample_TAGY.wov

fasta file with adapters/vector/tag cut; 1 file for each tag; 'TAGY' is the tags name that are cut from the sequences; Sequence codes are completed by '_A' if adapter is found at the beginning.
e.g.

```
>FVU26NR06DF571_MID1_A
ACCATTGCTTTGACTGACAGATGAATTGACATTACATTTCAGACAAACAAAAAGCCCCACATTCGCTC
TAAACACCCCTATCTGTCTCTGTCTCTCTGAAAACAGGCACATCCCACCTCAATAACAGATCAATCCC
GCCGACATTTGGACATTTATTCATTTTTCTCTCTCTCTCTCTCTTTCTCTGTCTCTGTTTCCTCTTTC
CCACTCAAAGAATGAAAACGAAATTAAACATTGAGCAAAAAGATAAATGGCGCCAACACGACAGCTCA
AAACACTCTCTCTGTTTATTGCTGAATG
```

Here the original sequence code is FVU26NR06DF571. The MID1 tag was removed from the sequence, as well as an adapter (_A)

6.1.2 sample_TAGY.woa

Sequences that did not have adapter at the beginning ('_N' is added to the sequence code).
e.g.

```
>FVU26NR06DF9P4_MID1_N
ATCTATCTATCTATCTATCTATCTATCCATTGCATTCTCATATCTTTGCTGTGTCATTCATTCATCTA
CCCATCCATTCAACATTTTTTTGGTCCATCCGTCAATCCATCCATCCATCCATCCATCCATCCATCCA
TGTATACTGTATCATACTTACATTTATCCATCCATGCATACTGCATTTCATTCATCATTCATCCATGC
ATGCTGT
```

In this sequence the MID1 tag was identified and removed, but no adapter was detected at the beginning of the sequence (_N)

6.1.2. sample_TAGY_length.tbl

text file with columns separated by ';'
Info on the number of bases cut from each sequence

Column1 Sequence code (e.g. FVU26NR06DF571_MID1_A)
Column2 Original length of the sequence (without tag) (e.g. 338)
Column3 Number of bases cut from the beginning of the sequence (e.g. 18)
Column4 Number of bases cut from the end of the sequence (e.g. 20)
Column5 Length of the sequence after cutting adapter/vector (e.g. 300)

```
FVU26NR06DF571_MID1_A;338;18;20;300;
FVU26NR06DF6CK_MID1_A;240;24;16;200;
FVU26NR06DF6HF_MID1_A;155;24;0;131;
```

6.1.2. sample_TAGY_80bp.seq

text file with column separated by ';'; info on MS motif and position

Column1: Sequence code
Column2: number of microsatellites in the sequence
Column3: length of the sequence
Column4: motif of the first microsatellite
Column5: first position of the microsatellite
Column6: last position of the microsatellite
Column7: number of repeats of the microsatellite
Colomns4-7 are repeated for all microsatellites

e.g.

```
FVU26NR06DF571_MID1_A;2;300;TC;164;179;8;CT;278;285;4;
```

2 microsatellites were found, both with TC motif. Positions of the first microsatellites are 164-179 (inclusive) and 278-285 for the second. The numbers of repeats are 8 and 4, respectively.

6.1.2. sample_pipe1.fas
1 file for each sample_TAGY.wov; Only sequences that have microsatellites, and are longer than the user defined limit.

6.2. Pipe2.pl

6.2.1. sample_pipe1_mask_redundant.tbl

Column1: Codes of redundant sequences (e.g. FVU26NR06DODXH_MID1_A)
Column2: Code of the longest sequence that covers completely the sequence in the first column with 100% identity (e.g. FVU26NR06DR449_MID1_A)
e.g.

```
FVU26NR06DODXH_MID1_A;FVU26NR06DR449_MID1_A
```

6.2.2. sample_pipe1_mask_nr.fas
Fasta file with all non-redundant sequences including original and concatenated sequences.

6.2.3. sample_pipe1_concat_info.txt

Column1: code of the concatenated sequence
Column2-3: code of the two sequences merged into one
e.g.

```
concat1;FVU26NR06DLPVR_MID2_A;FVU26NR06DODLB_MID2_A
```

The sequences FVU26NR06DLPVR_MID2_A and FVU26NR06DODLB_MID2_A have been merged into concat1

6.2.4. sample_pipe1_mask_nr_all_info.txt
info on each non-redundant sequence, fields are separated by ';', codes within fields are separated by ',‘

Column1: sequence code
Column2: sequences that were redundant to first sequence (100% identity) on the whole overlapping region
Column3: sequences where flanking region is identical, but MS is different
Column4: sequences with flanking region divergence is less than predefined limit (0.05 as default)
Column5: sequences similar to the first but flanking region divergence is more than predefined limit
Column6: sequences producing more than one hit with the query sequence (multihit)
Column7: Category (1) no BLAST hit to any other sequence (2) BLAST hit to a 100% identical sequence (3) BLAST hit to another sequence with identical flanking region, but difference in microsatellite (4) BLAST hit to another sequence with flanking region similarity over 95% (5) BLAST hit to another sequence with flanking region similarity less then 95%. Sequence with multihits are not taken into account.
e.g.

```
FVU26NR06DF8VH_MID2_A;;;;FVU26NR06DOWOQ_MID2_A;FVU26NR06DLYIG_MID2_A;5;
```

The sequence FVU26NR06DF8VH_MID2_A had significant blast hit to FVU26NR06DOWOQ_MID2_A but their similarity over the whole flanking region was less then 95%. FVU26NR06DOWOQ_MID2_A produced more than one blast hits to FVU26NR06DLYIG_MID2_A.

6.2.5. sample_pipe1_mask_nr_unique.fas
sequences that did not have BLAST hit or only to 100% identical sequences that had been removed previously.

6.2.6. sample_pipe1_mask_nr_groups.fas
groups of sequences (contigs) used to make consensus sequences
e.g.
16;FVU26NR06DNGX9_MID2_A,FVU26NR06DGB11_MID2_A,FVU26NR06DI8DX_MID2_A,
Conc16 is made based on the alignment of these three sequences.

6.2.7. sample_pipe1_mask_multihit_final.tbl
BLAST hits if there was more than one hit between a given sequence pair (probably minisatellites)
e.g.

```
FVU26NR06DLYIG_MID2_A   FVU26NR06DJAK4_MID2_A   96.12 103   4     0
      106   208   77    179   4e-045        172
FVU26NR06DLYIG_MID2_A   FVU26NR06DJAK4_MID2_A   91.54 130   8     1
      66    195   44    170   3e-043        167
```

The regions 106-208 and 66-195 of FVU26NR06DLYIG_MID2_A gave hit to the regions of 77-179 and 44 -170 of the FVU26NR06DJAK4_MID2_A.

6.2.8. sample_pipe2.fas

All original unique sequences plus consensus sequences that did not have blast hit.

6.3. Pipe3.pl

If ‘Select interrupted and compound microsatellites’ option is switched on, for each output file type there are two files: One with only perfect target microsatellites and one with only interrupted and compound target microsatellites.

‘compound’ in the filename refers to the files with compound and interrupted target microsatellites.

6.3.1. sample_pipe2_select**abc**.fas and sample_pipe2_compound_select**abc**.fas where,

a. The minimum length of the flanking region

b. The minimum length of the PCR product

c. The minimum number of repetitions for the target microsatellites

Fasta file with target microsatellites and sufficiently long nanosatellite free flanking region. All micro- and nanosatellites are printed in lowercase, and their positions and motif are written in the definition line of the sequence.

‘Pool’ as a motif means that at least two perfect nano-or microsatellites were pooled, because they were separated by two bp at most. In the definition line, all target microsatellite motifs are printed in lower case, and all nanosatellites are printed in uppercase letters. In this way, if you colour all lowercase letters, all nano- and microsatellites are easily spotted in the sequence, and all target microsatellite motifs are highlighted in the definition line.

e.g.

```
>FVU26NR06DF571_MID1_A POOL;81;96;2;pool;160;185;8;A;241;245;5;CT;278;285;4;
ACCATTGCTTTGACTGACAGATGAATTGACATTACATTTCAGACAAACAAAAAGCCCCA
CATTCGCTCTAAACACCCCTAtctgtctctgtctctcTGAAAACAGGCACATCCCACCTCAATAAC
AGATCAATCCCGCCGACATTTGGACATTTATTCAtttttctctctctctctctctttctcTGTCTCTGTTTCC
TCTTTCCCACTCAAAGAATGAAAACGAAATTAAACATTGAGCaaaaaGATAAATGGCGCC
AACACGACAGCTCAAAACActctctctGTTTATTGCTGAATG
```

The first and last positions of the target microsatellites are 160-185. It is a compound microsatellite, where the longest perfect microsatellite stretch is composed of 8 repeats.

6.3.2. sample_pipe2_ select**abc**_primer3.txt and sample_pipe2_compound_select**abc**_primer3.txt

Input file for Primer3. Target regions and excluded regions are given for each sequence. For detailed explanations see primer3 manual.

6.3.3. sample_pipe2_ select**abc**_tagret_ms_stat.tbl and sample_pipe2_compound_select**abc**_tagret_ms_stat.tbl

Text file with columns separated by ‘;‘

Column1: sequence code followed by a number to distinguish different target microsatellites of the same sequence (e.g. the first target microsatellites _0 of FVU26NR06DF571_MID1_A)

Column2: first position of the target microsatellite (e.g. 160)

Column3: length of the target microsatellite in base pairs (e.g. 26)

Column4: length of the target microsatellite in repeat numbers. For interrupted and compound microsatellites the highest number of repeats is given among the perfect di-hexabase motif microsatellites that make up the compound/interrupted microsatellite. (e.g. 8)

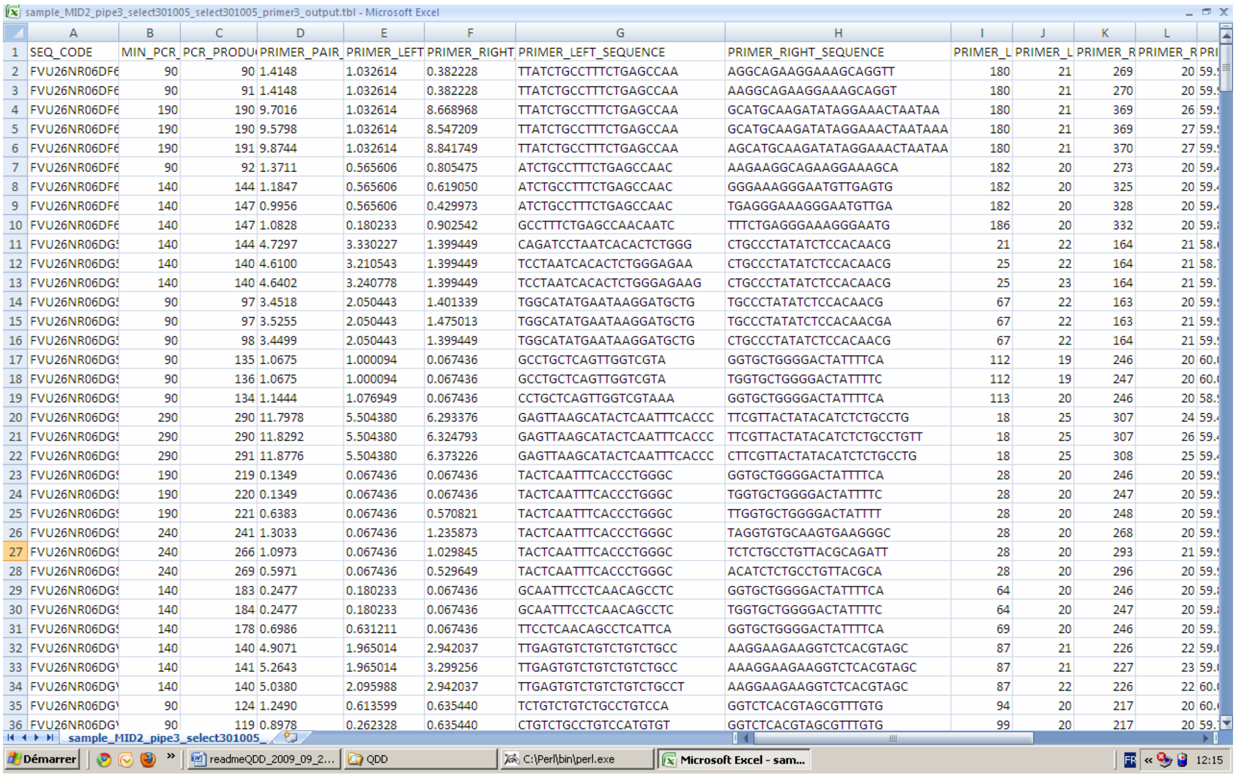
Column5: repeat motif for perfect microsatellites and the whole microsatellite for compound/interrupted microsatellite (e.g. TTTTTCTCTCTCTCTCTCTCTTTCTC)

Column6: transformed repeat motif (only for perfect microsatellites; AC is for AC, CA, TG, GT; AC is the first in alphabetical order among all motifs of the same group).

e.g.

```
FVU26NR06DF571_MID1_A_0;160;26;8;TTTTTCTCTCTCTCTCTCTCTTTCTC;
```

6.3.4. sample_pipe3_ abc_primers.tbl (THIS IS THE MOST IMPORTANT OUTFILE) and sample_pipe2_compound_abc_primers.tbl.



sample_MID2_pipe3_select301005_select301005_primer3_output.tbl - Microsoft Excel

	A	B	C	D	E	F	G	H	I	J	K	L	
1	SEQ_CODE	MIN_PCR_	PCR_PRODU	PRIMER_PAIR_	PRIMER_LEFT	PRIMER_RIGHT	PRIMER_LEFT_SEQUENCE	PRIMER_RIGHT_SEQUENCE	PRIMER_L	PRIMER_L	PRIMER_R	PRIMER_R	PRI
2	FVU26NR06DF6	90	90	1.4148	1.032614	0.382228	TTATCTGCCTTTCTGAGCCAA	AGGCAGAAGGAAAGCAGGTT	180	21	269	20	59.
3	FVU26NR06DF6	90	91	1.4148	1.032614	0.382228	TTATCTGCCTTTCTGAGCCAA	AAGGCAGAAGGAAAGCAGGT	180	21	270	20	59.
4	FVU26NR06DF6	190	190	9.7016	1.032614	8.668968	TTATCTGCCTTTCTGAGCCAA	GCATGCAAGATATAGGAAACTAATAA	180	21	369	26	59.
5	FVU26NR06DF6	190	190	9.5798	1.032614	8.547209	TTATCTGCCTTTCTGAGCCAA	GCATGCAAGATATAGGAAACTAATAAA	180	21	369	27	59.
6	FVU26NR06DF6	190	191	9.8744	1.032614	8.841749	TTATCTGCCTTTCTGAGCCAA	AGCATGCAAGATATAGGAAACTAATAA	180	21	370	27	59.
7	FVU26NR06DF6	90	92	1.3711	0.565606	0.805475	ATCTGCCTTTCTGAGCCAAC	AAGAAGGCAGAAGGAAAGCA	182	20	273	20	59.
8	FVU26NR06DF6	140	144	1.1847	0.565606	0.619050	ATCTGCCTTTCTGAGCCAAC	GGGAAAGGGAATGTTGAGTG	182	20	325	20	59.
9	FVU26NR06DF6	140	147	0.9956	0.565606	0.429973	ATCTGCCTTTCTGAGCCAAC	TGAGGGAAAGGGAATGTTGA	182	20	328	20	59.
10	FVU26NR06DF6	140	147	1.0828	0.180233	0.902542	GCCTTTCTGAGCCAACAATC	TTTCTGAGGGAAAGGGAATG	186	20	332	20	59.
11	FVU26NR06DG5	140	144	4.7297	3.330227	1.399449	CAGATCCTAATCACACTCTGGG	CTGCCCTATATCTCCACAACG	21	22	164	21	58.
12	FVU26NR06DG5	140	140	4.6100	3.210543	1.399449	TCCTAATCACACTCTGGGAGAA	CTGCCCTATATCTCCACAACG	25	22	164	21	58.
13	FVU26NR06DG5	140	140	4.6402	3.240778	1.399449	TCCTAATCACACTCTGGGAGAAG	CTGCCCTATATCTCCACAACG	25	23	164	21	59.
14	FVU26NR06DG5	90	97	3.4518	2.050443	1.401339	TGGCATATGAATAAGGATGCTG	TGCCCTATATCTCCACAACG	67	22	163	20	59.
15	FVU26NR06DG5	90	97	3.5255	2.050443	1.475013	TGGCATATGAATAAGGATGCTG	TGCCCTATATCTCCACAACGA	67	22	163	21	59.
16	FVU26NR06DG5	90	98	3.4499	2.050443	1.399449	TGGCATATGAATAAGGATGCTG	CTGCCCTATATCTCCACAACG	67	22	164	21	59.
17	FVU26NR06DG9	90	135	1.0675	1.000094	0.067436	GCCTGCTCAGTTGGTCGTA	GGTGCTGGGGACTATTTTCA	112	19	246	20	60.
18	FVU26NR06DG9	90	136	1.0675	1.000094	0.067436	GCCTGCTCAGTTGGTCGTA	TGGTGCTGGGGACTATTTTC	112	19	247	20	60.
19	FVU26NR06DG9	90	134	1.1444	1.076949	0.067436	CCTGCTCAGTTGGTCGTAAA	GGTGCTGGGGACTATTTTCA	113	20	246	20	58.
20	FVU26NR06DG9	290	290	11.7978	5.504380	6.293376	GAGTTAAGCATACTCAATTTCACCC	TTCGTTACTATACATCTCTGCCTG	18	25	307	24	59.
21	FVU26NR06DG9	290	290	11.8292	5.504380	6.324793	GAGTTAAGCATACTCAATTTCACCC	TTCGTTACTATACATCTCTGCCTGTT	18	25	307	26	59.
22	FVU26NR06DG9	290	291	11.8776	5.504380	6.373226	GAGTTAAGCATACTCAATTTCACCC	CTTCGTTACTATACATCTCTGCCTG	18	25	308	25	59.
23	FVU26NR06DG9	190	219	0.1349	0.067436	0.067436	TACTCAATTTCACCCTGGGC	GGTGCTGGGGACTATTTTCA	28	20	246	20	59.
24	FVU26NR06DG9	190	220	0.1349	0.067436	0.067436	TACTCAATTTCACCCTGGGC	TGGTGCTGGGGACTATTTTC	28	20	247	20	59.
25	FVU26NR06DG9	190	221	0.6383	0.067436	0.570821	TACTCAATTTCACCCTGGGC	TTGGTGCTGGGGACTATTTT	28	20	248	20	59.
26	FVU26NR06DG9	240	241	1.3033	0.067436	1.235873	TACTCAATTTCACCCTGGGC	TAGGTGTGCAAGTGAAGGGC	28	20	268	20	59.
27	FVU26NR06DG9	240	266	1.0973	0.067436	1.029845	TACTCAATTTCACCCTGGGC	TCTCTGCCTGTTACGCAGATT	28	20	293	21	59.
28	FVU26NR06DG9	240	269	0.5971	0.067436	0.529649	TACTCAATTTCACCCTGGGC	ACATCTCTGCCTGTTACGCA	28	20	296	20	59.
29	FVU26NR06DG9	140	183	0.2477	0.180233	0.067436	GCAATTTCCTCAACAGCCTC	GGTGCTGGGGACTATTTTCA	64	20	246	20	59.
30	FVU26NR06DG9	140	184	0.2477	0.180233	0.067436	GCAATTTCCTCAACAGCCTC	TGGTGCTGGGGACTATTTTC	64	20	247	20	59.
31	FVU26NR06DG9	140	178	0.6986	0.631211	0.067436	TTCCTCAACAGCCTCATTCA	GGTGCTGGGGACTATTTTCA	69	20	246	20	59.
32	FVU26NR06DG	140	140	4.9071	1.965014	2.942037	TTGAGTGTCTGTCTGTCTGCC	AAGGAAGAAGGTCTCACGTAGC	87	21	226	22	59.
33	FVU26NR06DG	140	141	5.2643	1.965014	3.299256	TTGAGTGTCTGTCTGTCTGCC	AAAGGAAGAAGGTCTCACGTAGC	87	21	227	23	59.
34	FVU26NR06DG	140	140	5.0380	2.095988	2.942037	TTGAGTGTCTGTCTGTCTGCCT	AAGGAAGAAGGTCTCACGTAGC	87	22	226	22	60.
35	FVU26NR06DG	90	124	1.2490	0.613599	0.635440	TCTGTCTGTCTGCCTGTCCA	GGTCTCACGTAGCGTTTGTG	94	20	217	20	60.
36	FVU26NR06DG	90	119	0.8978	0.262328	0.635440	CTGTCTGCCTGTCCATGTGT	GGTCTCACGTAGCGTTTGTG	99	20	217	20	59.

sample_MID2_pipe3_select301005_

Text file with columns separated by ';'

Column1: sequence code followed by a number to distinguish different target microsatellites of the same sequence

Column2: The minimum length of the PCR product set for Primer3

Column3-24 see documentation of Primer3

Column25-29: same as column 2-6 of sample_pipe2_selectabc_tagret_ms_stat.tbl

Column30: 'best' if the primer pair has the lowest penalty for a given marker, otherwise empty. The number of lines marked as best gives the number of markers for which Primer3 designed at least 1 primer pair. When opening this file in excel, do not forget to replace '.' by ',' if your excel uses ',' to separate integers from decimal values. All sequence codes that starts with 'cons' are consensus sequences (*e.g.* in 'cons12_A_3', cons12 is an identifier, A means that all sequences in the contig had a vector removed from the beginning, and there were 3 sequences in the contig. Concatenated sequences (with 100% identity in the overlapping region) have an identifier beginning by 'concat'. All other identifiers refer to original sequences.

6.3.5. 'html' and 'html_compound' subfolders of sample_TagX_pipe3 contain index.html, which is the html equivalent of sample_pipe3_ abc_primers.tbl or sample_pipe2_compound_abc_primers.tbl. It contains an additional column at the beginning with the sequence codes in hyperlink. By clicking on these hyperlink the fasta file of a given sequence can be opened by the editor of your choice.

7. Troubleshooting

7.1. Pipeline 1 starts but the window closes immediately

→ Check that adapter.fas and tag.fas are located in the input folder (if options of adapter removal and tag sorting are switched on)

→ Do not run the perl script by clicking on the filename in explorer, but use the clean way of opening a terminal (5.1.2).

7.2. Pipeline 2 produces empty consensus alignments

→ Make sure Clustal 2 is installed in the folder chosen at installation using the msi program.

7.3. Read access to some files is refused

→ Make sure you run only one perl script at a time

8. Reference

Emese Meglécz, Caroline Costedoat, Vincent Dubut, André Gilles, Thibaut Malausa, Nicolas Pech and Jean-François Martin 2009. QDD: a user-friendly program to select microsatellite markers and design primers from large sequencing projects. **Bioinformatics** (Accepted)