Fachhochschule
University of Applied Sciences
Oldenburg
Ostfriesland
Wilhelmshaven

AWI

Extension of a sequence analysis system for multiplex primer design

Student: Lars Kraemer

Matriculation number: 5001115

Contact: Lars.Kraemer@awi.de

Study course: Bioinformatics

Semester: 12 (SS 2008)

Company: Alfred-Wegener-Institut Bremerhaven

Sojourn time: 1. January 2008 – 1. Mai 2008

First examiner: Prof. Dr. rer. nat. Gerhard Kauer

Second examiner: Prof. Dr. Stephan Frickenhaus

Statutory Declaration

I explain the fact that I wrote this work independently and used no other sources or aids than those indicated.

Bremerhaven, 17. May 2008 *Lars Kraemer*

Zusammenfassung der Arbeit:

Das Projekt erlaubt die Detektion von Mikrosatelliten, zur Verwendung als genetische Marker. Weiterhin ermöglicht es das Design von PCR Primern für diese Marker und die Zusammenstellung dieser Primer zur Verwendung in Multiplex PCR Experimenten. Für eine einfache Handhabung der erzeugten Daten können diese in einer Datenbank gespeichert werden.

Um dies zu erreichen, wurden Programme zur Mikrosatelliten Erkennung (Phobos) und dem Design von Primern (Primer3) in eine Sequenz Analyse Software, dem Staden Package, integriert. Dies geschah durch Tcl Skripte. Auf die gleiche Weise wurde ein Modul erstellt, welches die Zusammenstellung von Primerpaaren für die Verwendung in Multiplex PCR Experimenten erlaubt. Die Datenbank Eigenschaft wurde durch SQLite ermöglicht, welches bereits über eine Tcl Einbindung verfügt.

Das Dokument beschreibt die funktionsweise und die Handhabung der erzeugten Tcl Skripte und schafft einen Überblick über die verwendeten Programme und Technologien, die zum erreichen der Funktionalität des Projekts verwendet wurden. Des Weiteren enthält das Dokument Ergebnisse von Laborexperimenten, welche die Funktionsfähigkeit der entwickelten Module untermauern. Aufgrund von Zeitmangel konnte nur in funktionierender Multiplex, bestehend aus zwei Primerpaaren, dokumentiert werden.

Zum Abschluss lässt sich sagen, dass das Projekt eines der besten, im Moment verfügbaren Werkzeuge zum Design von Mikrosatelliten Markern ist.

Summary of the project:

The project provides workflows for a fast detection of microsatellites as genetic markers, the creation of PCR primers for these markers and composing these primers for usage in multiplex PCR experiments. All the produced data can be stored in a database for easy handling.

To achieve this, tools for microsatellite detection (Phobos) and primer design (Primer3) were integrated into the sequence analysis software, called Staden package. The integration happened through Tcl scripts, which were also used for the development of a module for primer selection for usage in multiplex PCR experiments. The database feature was established with SQLite, which possesses already Tcl bindings.

This document describes the mode of operation and the handling of the developed Tcl scripts and outlines the used tools for achieving their functionality. Furthermore, it includes results of laboratory PCR experiments to proof the functionality of the created workflows. Through lack of time only allowed to document one multiplex consisting of two primer pairs.

Finally, it can be said that the project is one of the best microsatellite marker design tool available at the moment.

1 Introduction

Microsatellites are widely used molecular markers, e.g., in population genetics they serve as a tool to characterize different strains of a species and retrace their spreading or define the interaction of different groups of a species. This is possible through polymorphic microsatellites (MS) and their amplification with polymerase chain reaction (PCR) [Mullis, 1983]. The PCR products show differences between the lengths of polymorphic MS of different groups of a species. The number of different MS serves as a hint to the relation of these groups. The design of PCR primers for the genotyping of microsatellites is a time consuming task which can be greatly facilitated by integrated analysis software. Furthermore, the application of these primers could be less laborious through usage of multiplex-PCRs. Also for this task of composing primer pairs useable in a multiplex-PCR simultaneously, analysis software provides simple and time saving applications.

With the aid of existing components (Staden Package, Phobos and Primer3) exemplary workflows shall be constructed, which allow the automatic design of genetic markers from shotgun-sequences. Special emphasis is put on the level of integration, i.e., the workflow is completely integrated into one framework, making use of existing and new database structures.

The Windows version of the Staden-Package provides a basis, in which, through Tcl-scripts (Tool command language) [Ousterhout, 1988], further components, called *modules*, shall be included.

The goals of this project were the development and improvement of extensions for the Staden Package. These module extensions should allow the identification of short repeated sequences, the design of primers flanking these repetitive regions, composing these primers for usage in a multiplex-PCR and creating a database to store and handle the created data together with primary data, i.e., trace files from a sequencer.

Altogether this project shall provide workflows for a fast detection of microsatellites as genetic markers, the creation of PCR primers for these markers and composing these primers for usage in multiplex-PCR experiments. All this is supposed to be stored in a database for an easy data handling.

2 Biological background

2.1 Microsatellites

Microsatellites or SSR (Simple Sequence Repeats) are short repetitive DNA-Sequences which appear in the genomes of all organisms [Jarne and Lagoda, 1996]. The repeat sequence of a SSR consists mostly of 2-4 nucleotides [Turnpenny and Ellard, 2005] and will be repeated 10- to 100-times.

Both quantity and length of the microsatellites in a DNA-segment can vary from individual to individual. They posses a high analytical power (multi locus, multi-allelic, co-dominant, nuclear, single-copy markers; [Schlötterer, 2004]); hence they became one of the most widely used molecular marker systems.

Genotyping of microsatellite loci is usually performed by PCR (Polymerase Chain Reaction) amplification and subsequent length determination by polyacrylamide gel electrophoresis or on automatic DNA sequencers. Precondition for this is the knowledge of the sequence of bases flanking the microsatellite which shall serve as a marker. These flanking regions provide the basis for the design of oligonucleotide-primers responsible for the specificity of the amplification.

Developed microsatellites primers for a particular species can also be applied to closely related species. The number of successfully amplified loci will decrease by increasing genetic distance [Jarne and Lagoda, 1996]. This happens, e.g., through point mutations within primer regions which prevent the amplification of affected loci.

2.2 Microsatellite marker design

The design of microsatellite markers begins with the DNA-sequences of the examined organism. Normally the sequences are available as numerous trace files which represent short fragments of the DNA flanked by vector sequences. These fragments originate from a genomic library and were digitalised with a DNA-sequencer.

The sequences should overlap each other to allow a reassembling of the original DNA-strand. In practice a complete reassembling of the DNA-strand is not ensured. It depends on the quality of the sequencer data, which decreases with the length of the fragment. It is also not uncommon that a fragment ends in the middle of a microsatellite, which provokes wrong overlaps.

In order to allow a decent assembly the accuracy of the bases in the sequences has to be ensured and the vector regions in the fragments have to be cut off, otherwise they would lead to wrong overlaps. In addition the microsatellites in all sequences, especially at the beginning or end of a sequence, have to be masked.

After the DNA-sequences are assembled the search for usable repeats can be started. Normally di-repeats are picked for the marker design. When the microsatellites are found, primers have to be created in the flanking regions of these repeats. Here it is necessary to look for other repeats between the primer pairs, because length variations in those repeats could cause wrong results if the primer pair was used as marker. Sometimes these microsatellites are hard to find, e.g., they could be imperfect or have a repeat unit of five or six nucleotides.

If the created primer pairs are supposed to be used in a multiplex-PCR experiment it must be checked that the various primers will not interact with each other or have multiple binding sites in the sample DNA. Furthermore, all primers ideally share nearly the same melting / annealing temperature.

3 Used technologies

3.1 Script-language Tcl

Tcl (**T**ool **c**ommand **l**anguage, developed by John Ousterhout, 1988) is an open source script-language and related to Unix-shell-languages. It is easy to learn and allows application development in short time.

Tcl is well known in the toolkit Tk. Tk is an open source toolkit for development of graphical user interfaces (GUIs). It was created for Tcl and is usable for small projects under Linux, Windows or Macintosh. Tk-implementations are also available for other languages like Perl or Python.

3.2 The Staden package

The Staden package is a sequence handling and analysis software developed at the Medical Research Council Laboratory of Molecular Biology in Cambridge, UK.

Download: *https://sourceforge.net/projects/staden/*

It is based on C methods which are accessed through Tcl-scripts.

Two components of the Staden-Package are of primary interest to the present project as described in 3.2.1 and 3.2.2.

3.2.1 Pregap4

Through a graphical user interface trace data can be prepared for assembly or analysis. Automated processes are for example base calling, trace format conversion, quality analysis, vector clipping, contaminant screening, repeat searching and mutation detection.

3.2.1.1 Modules

Pregap4 is an interface which allows processing any kind of sequence data through the execution of optional and configurable modules in a step-by-step scheme as a workflow. It allows an easy integration of new modules into its structure.

3.2.1.1.1 Phred

Phred is a C program for the base calling of DNA sequencer trace files in SCF, ABI or ESD format [Ewing & Green, 1998]. The base calls and assigned quality values are written to output files. Phred uses simple Fourier methods for the trace examination. In Pregap4 it can be used for exactly this purpose.

More information can be found on the web at *http://www.phrap.com/phred/*.

3.2.1.1.2 Crossmatch

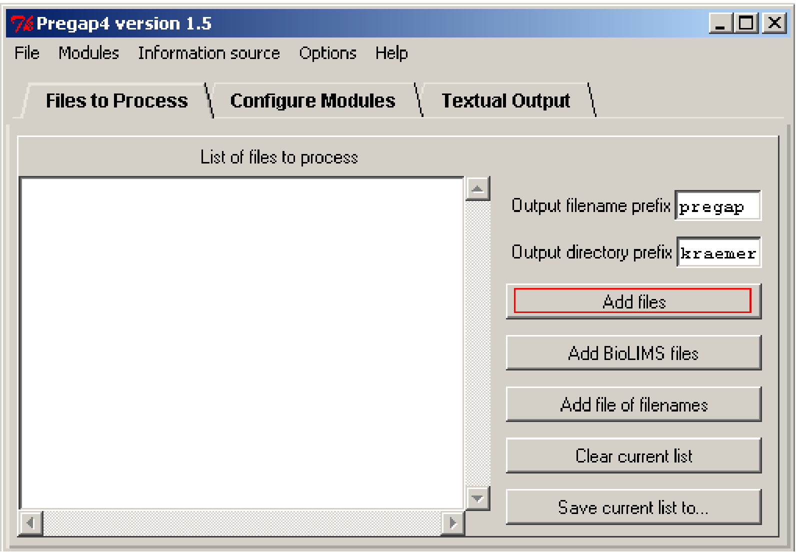
Crossmatch is a program for comparing nucleic acid or protein sequences. It is based on the Smith-Waterman-Gotoh algorithm [Smith & Waterman 1981, Gotoh 1982]. In comparison to BLAST, which is a heuristic alignment program, it is slower but more sensitive, providing globally optimal alignments by means of dynamic programming. In Pregap4 it is used to compare vector sequences against reads and mask the vector sequence regions.

3.2.1.1.3 Phobos

The Phobos module for Pregap4 was invented in an earlier project [Kraemer, 2007], but some methods were rewritten or completely removed because of three reasons: a new Phobos version, a changed tag appearance and a better comprehensibility of the script. The module allows the setting of the motif-length and minimum number of motif repetitions of the SSRs, which are found by Phobos. The microsatellites which fit to the set parameters are written as tags to experiment files. These files were created by Pregap4 from the trace data through another module.

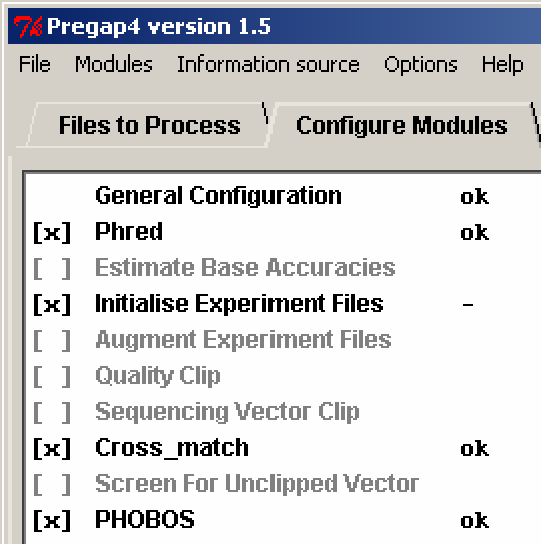
3.2.1.2 Processing trace files with Pregap4

To process trace files with Pregap4, they have to be loaded into the “Files to Process”-table through the “Add files”-button (see the picture below).



Output filename prefix pregap

Output directory prefix kraemer



```
    General Configuration            ok
[x] Phred                            ok
[ ] Estimate Base Accuracies
[x] Initialise Experiment Files      -
[ ] Augment Experiment Files
[ ] Quality Clip
[ ] Sequencing Vector Clip
[x] Cross_match                      ok
[ ] Screen For Unclipped Vector
[x] PHOBOS                           ok
```

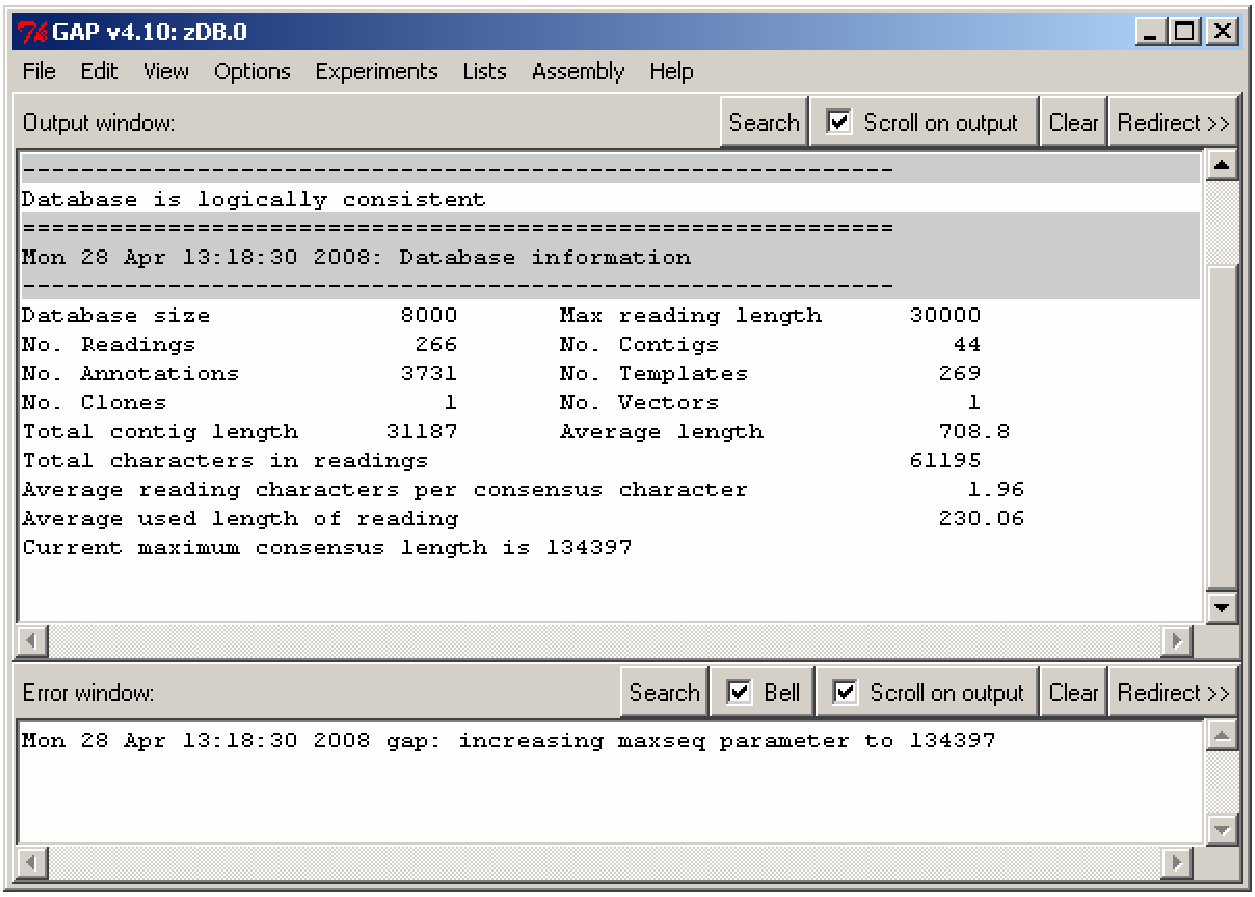
When the trace files are selected, the modules which shall be used have to be selected and eventually loaded. These settings are done in the “Configure Modules”-table.

For creating experiment files with marked repeats for a Gap4 database, the modules listed in the menu displayed here were used in the shown order. Possibly the Phred and Crossmatch modules have to be installed and the Phobos module must be loaded. Loading a module is possible through the “Modules”-menu and the option “Add/Remove Modules”. Crossmatch needs a FASTA file with the vector sequence of the trace files. Instead of Crossmatch the “Sequencing Vector Clip”-module could be used. The other unselected modules in the menu could be used together with the selected ones in the shown order, but they are not necessarily needed.

After selecting and, if necessary, configuring the modules the “Run”-button at the left bottom of the Pregap4 window is to be pressed to initiate the processing of the modules. All created files in this process are written to the file-folder of the loaded trace files.

3.2.2 Gap4

Gap4 allows the creation of a database and the assembly of sequences, ordering and joining contigs based on read pair data and sequence comparisons. It is possible to analyze and edit the contigs. Also it gives a graphical view of assembled sequences. Here the graphical user interface to Gap4 is displayed after loading a database.



```
------------------------------------------------------------
Database is logically consistent
============================================================
Mon 28 Apr 13:18:30 2008: Database information
------------------------------------------------------------
Database size                 8000       Max reading length      30000
No. Readings                   266       No. Contigs                44
No. Annotations               3731       No. Templates             269
No. Clones                       1       No. Vectors                 1
Total contig length          31187       Average length          708.8
Total characters in readings                                     61195
Average reading characters per consensus character                 1.96
Average used length of reading                                   230.06
Current maximum consensus length is 134397
```

```
Mon 28 Apr 13:18:30 2008 gap: increasing maxseq parameter to 134397
```

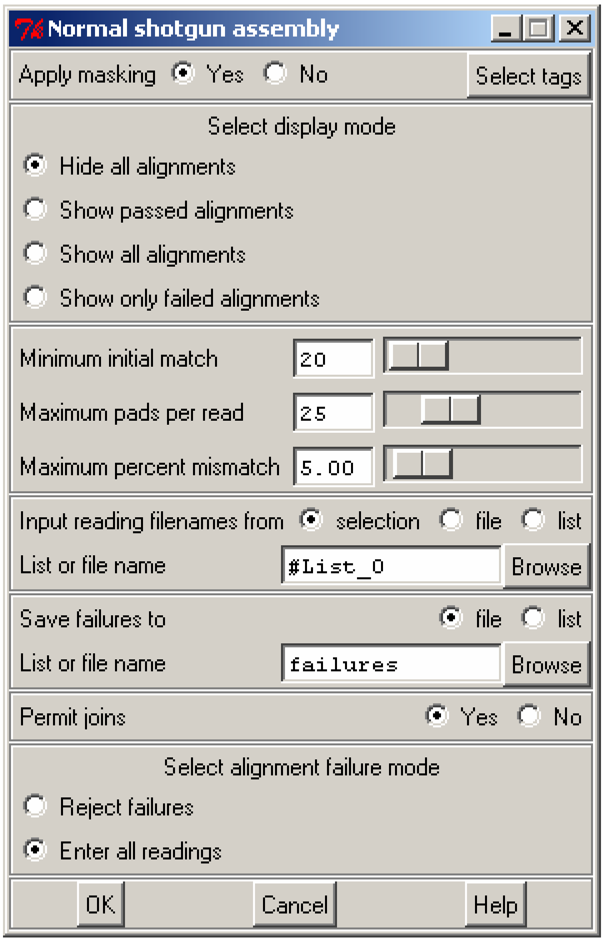
The Gap4 manual can be found on the following webpage:

http://staden.sourceforge.net/manual/gap4_unix_toc.html

3.2.2.1 Creating a Gap4 database

At first a new and empty Gap4 database has to be created. This is done through the "File"-menu and a click on the "New"-option. With the "Browse"-button of the opened window the right folder for the database can be chosen. It should be the same folder as the one with the experiment-files which will be included in the database; otherwise it will not be possible to write something to the files, only to the database.

Now an empty database exists. To fill it with some sequence-data the "Assembly"-menu gets selected and the kind of assembly, which shall be performed, is chosen. The "Normal shotgun assembly" was used in this project. The assembly parameters were set like on the picture to the left.



Normal shotgun assembly

Apply masking ● Yes ○ No

Select tags

Select display mode

● Hide all alignments
○ Show passed alignments
○ Show all alignments
○ Show only failed alignments

Minimum initial match 20

Maximum pads per read 25

Maximum percent mismatch 5.00

Input reading filenames from ● selection ○ file ○ list

List or file name #List_0 Browse

Save failures to ● file ○ list

List or file name failures Browse

Permit joins ● Yes ○ No

Select alignment failure mode

○ Reject failures
● Enter all readings

For the masking the "SVEC" and "REPT"-tags were selected, through this an assembly by means of overlapping microsatellites were prevented. Under the option "Input reading filenames from" the experiment-files for the database are chosen. This happens best with the radio-button on "selection" and browsing for the files. It is necessary that all files are in the same folder. Furthermore the path to the folder and the file names are not allowed to include space characters, also there is a limit for the length of file names. Under "Save failures to" a filename has to be set. After clicking the "OK"-button the sequences will be assembled and written to the database.

3.3 BLAST

BLAST (**B**asic **L**ocal **A**lignment **S**earch **T**ool) stands for a collection of tools for the analysis of sequence data [Altschul *et al*, 1990]. In this project "blastall" from the NCBI blast was used. It compares DNA or protein sequences to sequences already in a database. The result is a list of local alignments with scores quantifying the significance of the different alignment hits.

BLAST can be downloaded at *http://www.ncbi.nlm.nih.gov/blast/download.shtml*.

3.4 Phobos

Phobos was developed by Christoph Mayer [2008] (Univ. Bochum) for microsatellite detection. Phobos uses files of DNA-sequences in FASTA-format as input. As result it returns a text file with a list of found SSRs for each sequence. Phobos is able to find imperfect repeats and can search for SSRs with a repeat length much longer than six nucleotides. Also it can be flexibly customized considering for example the imperfection of repeats to be found.

A list of the Phobos parameters and their function can be found in "The Staden Multiplex Project.pdf" in the doc-folder of the Staden Package after installing the project.

Download: *http://www.ruhr-uni-bochum.de/spezzoo/cm/cm_phobos.htm*.

3.5 Primer3

Primer3 is a command line based program [Rozen & Skaletsky, 2000]. Its purpose is to find possible DNA-primer positions in a DNA-sequence, normally for given target regions in the sequence. The input file must have a special format with the DNA-sequence, eventually quality data and the specifics for the primer search. Primer3 returns the results of the search as text-output to standard output, gathering all information about the found primer pairs.

3.6 SQLite

SQLite is a program library with an integrated relational database system; it was developed by Dr. Richard Hipp [2000] and is written in C. The database is designed for embedded usage, i.e., without client-server concepts; it is available with interfaces for all important programming languages including Tcl. Through the embedded design the application can be integrated into programs without the need for additional server software.

The SQLite library is available under *http://www.sqlite.org/*.

3.7 The R-Project

The statistics software R was released under the GNU licence. It is available for Windows, Linux, MacOS and further operating systems. It was developed by Ross Ihaka and Robert Gentleman and is based on the S and S-Plus language.

R provides many statistical and graphical techniques; also it can be easily enhanced by a variety of publicly available packages. Many of these packages are directly available through the R-Project, see *http://www.r-project.org.*

4 The project

4.1 Approach

The project should establish an easy and fast way to create microsatellite markers for multiplex-PCR experiments from trace files. Since the Staden Package delivered a good workflow for sequence analysis through Pregap4 and Gap4 and an earlier project already integrated modules for microsatellite detection and primer creation [Kraemer, 2007] in the Staden Package, it was the optimal choice as basis for this project.

The work started directly with the Multiplex module, because there were already modules which allowed the creation of primers for microsatellites. The Multiplex module should be able to compare huge lists of primer pairs against each other to eliminate possibly conflicting primer combinations. Furthermore, the module should allow comparing the primers against the sequences in the Gap4 database and excluding primer pairs showing similarities to non-target sequences.

For comparing the primers against the database, NCBI BLAST (see section 3.4) was integrated in the module. The module can be configured to blast primers against the sequences from the Gap4 database, or against any other user-specified set of sequences. Primers with good hits can be excluded from further comparisons as probably unspecific.

For the comparison of the primers a modified version of “ntdpal”, a component of Primer3, was used (see section 4.2.7). In Primer3 “ntdpal” is used to compare one primer against a second primer. It returns a score value for their compatibility. The modified version allows comparing whole primer lists against each other and it returns a matrix of the score values.

Finally, an algorithm for selecting primer pairs for a multiplex-PCR experiment based on the score matrix, the blast results and a manual selection option was implemented.

Besides, a comfortable way for storing and handling of the created primer lists had to be developed to circumvent inconvenient processing of files. At the moment the lists only could be written to the Gap4 database, the experiment files or be exported as text files.

SQLite seemed a promising option, because of the already existing Tcl-bindings, at least for Windows and Linux.

The SQLite module for Gap4 was not only supposed to allow the storing of lists, but also to reuse them, or selected parts of them, as input for modules, e.g., a repeat list as input for the Primer3 module. To make this possible the already existing modules for microsatellite detection and primer creation had to be considerably changed.

The Phobos module was nearly completely rewritten. Originally it only detected repeats and wrote them to the Gap4 database. Now it allows creating multiple repeat lists with different contents, displayed in separate windows. Here every single repeat can be selected individually and stored not only in the Gap4 database but also in a SQLite database or exported as text file. This considerably improves the flexibility of the software. Furthermore, the appearances of the Gap4 database and experiment file tags were changed.

Similar modifications were made for the Primer3 module. The tag modifications led also to further developments of the Phobos module for Pregap4.

4.2 Design and implementation

4.2.1 phobos.p4m

Phobos.p4m is the Phobos module for Pregap4. The module based on the Troll module for Pregap4 [Martins *et al*, 2006]. For this project it was nearly completely rewritten. Methods no longer needed were removed and the tag marking of experiment files was simplified and adjusted to the needs of the new Gap4 modules.

The methods:

configure_dialog

This method is invoked by Pregap4 when the module configuration shall be saved. The current configuration of the modules interface variables is written to a Pregap4 config file.

create_dialog

The method creates the interface of the Phobos module.

init

By processing the module through Pregap4 the method checks for the “phobos_cl”-file in the binaries folder of the Staden Package, if it is not present the module cannot be run and throws an error message.

name

This method returns the module name "PHOBOS" to Pregap4.

process_dialog

It gathers all variable setting from the module interface.

run

This method contains the whole processing part. At first it creates a temporary FASTA file with all sequences of the input experiment files. This file serves as input for Phobos. After Phobos is executed the results are written to the "pregap.Phobos_output"-file. This file will now be used to create repeat tags in the original input experiment files.

4.2.2 gap_phobos.tcl

The earlier version of the module only was able to find SSRs through Phobos and write them to the Gap4 database or the experiment files.

The module was enhanced by a table view of the found repeats and through this gained access to the contig position of every repeat. The option to store the repeat data in a SQLite database was added. Furthermore some bugs were fixed and some methods rewritten. All in all the comfort and flexibility of the module was much enhanced.

The methods:

gap_phobos

This method is executed by invoking the module. It creates the interface of the Phobos module.

gap_phobos_close

The method is invoked through the "Close Window"-button in the interface. It creates the "Phobos_SSR.passed" and "Phobos_no_SSR.passed" files and deletes the Phobos output file, depending on the settings in the interface.

gap_phobos_editEndCmd

It is linked to the tablelist object and is invoked by clicking on the "Use"-column of a row in the tablelist. It allows changing the state of the checkbox in this column.

gap_phobos_execute

This will be invoked through the button "*Run Phobos*" in the Phobos interface. It creates a text file with all consensus sequences and tags of the contigs from the database. From this a FASTA file, as Phobos input, is created. After that Phobos is executed with the input file. The input and consensus files get deleted and the Phobos results are written to a text file.

gap_phobos_export_repeats

This method is executed through the "Export to File"-button in the "Repeat Table"-window and writes all, through the "Use"-checkbox, selected repeats to a text file. The text file has the name set in the "Export to File"-box.

gap_phobos_export_write

It is invoked through the "*gap_phobos_export_repeats*"-method, when the filename already exists. It opens a dialog which allows overwriting the existing file, to append the repeats to the existing file or to abort the writing progress.

gap_phobos_generate_table

It is executed through the button "Generate Table" in the Phobos interface. The method "*gap_phobos_get_results*" is invoked and the returned results will be search for fitting repeats on basis of the parameters set in the interface. The list with the selected SSRs is given to the method "*gap_phobos_table*".

gap_phobos_get_results

This method gets the microsatellite information from the Phobos output file and stores them in a list, which is returned to the "*gap_phobos_generate_table*"-method. Because the Gap4 database contigs can include more than one reading, it is necessary to find the right position in the database for the repeats before they can be entered into the result list.

gap_phobos_invoke_editors

The method is linked to the tablelist object and invokes a contig-editor through a double click on a row in the tablelist. The contig-editor will show the location of the selected repeat.

gap_phobos_populate_tablelist

It will be executed through the buttons “Select All” or “Deselect All” in the tablelist. It clears the repeat table and fills it anew with the same repeats, but with different state of the “Use”-column of the table.

gap_phobos_primer3

It is invoked through the “Create Primer”-button in the “Repeat Table”-window. The method creates a target list of the selected repeat rows in the tablelist and then invokes an instance of the Primer3 module with this target list for primer creation.

gap_phobos_save

This method is processed after pressing the button “Save” in the Phobos interface. It saves the parameter settings of the interface to a file in the “phobos_multiplex”-folder of the Staden Package. This file will be loaded automatically with every new start of the module.

gap_phobos_sqLiteDB

This method is invoked through the “Export to SQLiteDB”-button in the “Repeat Table”-window. It opens or creates a SQLite database with the name set in the “Export to SQLiteDB name”-box in the window. It will enter the selected rows as a new list in the SQLite database. To verify that the new table is not already in the database an md5sum of the list is created and compared to the lists already in the database.

gap_phobos_table

This method creates the tablelist window. It gets a list of repeats from the method “*gap_phobos_generate_table*” and enters them to the table.

gap_phobos_writeToExpOrDB

By pressing the “Add to”-button, depending on the status of the “DB”- and “exp-File”-checkboxes, this method will add the selected repeats as tags to the Gap4 database or/and the experiment files.

md5sum

It is executed through the method “gap_phobos_sqLiteDB” and creates an md5sum of a given list.

4.2.3 gap_primer3.tcl

The earlier version of the module was able to create flanking primers for every tag type in a Gap4 database, for example microsatellites. The created primer pairs could be written as tags to the Gap4 database or the experiment files, also it was possible to export the primers as tab-delimited text file.

Beside some fixes and method rewriting, the module now allows multiple instances and is able to write the results to a SQLite database. Furthermore it is able to create flanking primers for repeats not only from the Gap4 database, but also from a SQLite database.

The methods:

gap_primer3

This method is executed by invoking the module. It creates the interface of the Primer3 module.

gap_primer3_create_primer_list

This method gets the primer information from the Primer3 output file and stores them in a list, which is returned to the "*gap_primer3_tablelist*"-method. Because the Gap4 database contigs can include more than one reading, it is necessary to find the right position in the database for the repeats before they can be entered into the result list. Normally, for every microsatellite with proper flanking regions for primers, four primer pairs are created. Only the best primer pair for a microsatellite will later be selected in the tablelist.

gap_primer3_editEndCmd

It is linked to the tablelist object and is invoked by clicking on the "Use"-column of a row in the tablelist. It allows changing the state of the checkbox in this column.

gap_primer3_execute

This will be invoked at the end of the method "*gap_primer3_ok*". It executes Primer3 with the input file. The Primer3 results are written to a text file.

gap_primer3_export_primer

This method is executed through the "Export to File"-button in the "Primer Table"-window and writes all, through the "Use"-checkbox, selected primer pairs to a text file. The text file has the name set in the "Export to File"-box.

gap_primer3_export_write

It is invoked through the "*gap_primer3_export_primer*"-method, when the filename already exists. It opens a dialog which allows overwriting the existing file, to append the primer pairs to the existing file or to abort the writing progress.

gap_primer3_get_targets

Invoked through the "*gap_primer3_ok*"-method it creates the lists of target- and excluded-regions for the Primer3 input file. Depending on the setting of the parameters for gap-length, target-length and the "Only single SSR"-mode the method decides if a possible target region is entered to the target list or not. If there are not enough bases between two SSRs to place a primer, the SSRs do not get on the target list or will be counted as a single target, depending on the "Only single SSR"-mode.

gap_primer3_invoke_editors

The method is linked to the tablelist object and invokes a contig-editor through a double click on a row in the tablelist. The contig-editor will show the location of the selected primer pair product graphically.

gap_primer3_mark_primer

By pressing the "Add to"-button, depending on the status of the "DB"- and "exp-File"-checkboxes, in the "Primer Table"-window, this method will add the selected primer pairs as tags to the Gap4 database or/and the experiment files. For marking the primers in the Gap4 database the right position in the right reading of the given contig has to be found.

gap_primer3_multiplex

It is invoked through the "Create Multiplex"-button in the "Primer Table"-window. The method creates a primer list of the selected primer pairs in the tablelist and then invokes an instance of the Multiplex module with this primer list for multiplex creation.

gap_primer3_ok

This method is invoked when the "OK"-button of the Primer3 interface is pressed. It creates a text file with the consensus sequences and tags of all contigs. With this file and the method "*gap_primer3_get_targets*" a Primer3 input file is created. After that it invokes the method "*gap_primer3_execute*". At last the method "*gap_primer3_create_primer_list*" is processed.

gap_primer3_populate_tablelist

It will be executed through the buttons “Select All”, “Deselect All” or “Standard Selection” in the tablelist. It empties the primer table and fills it anew with the same primer pairs, but with different state of the “Use”-column of the table. The “Standard Selection” marks, for every repeat, only the primer pair with the lowest penalty value.

gap_primer3_safe

This method is processed after pressing the button “Save” in the Primer3 interface. It saves the parameter settings of the interface to a file in the “phobos_multiplex”-folder of the Staden Package. This file will be loaded automatically with every new start of the module.

gap_primer3_sqLiteDB

This method is invoked through the “Export to SQLiteDB”-button in the “Primer Table”-window. It opens or creates a SQLite database with the name set in the “Export to SqLiteDB name”-box in the window. It will enter the selected rows as a new list in the SQLite database. To verify that the new table is not already in the database an md5sum of the list is created and compared to the lists already in the database.

gap_primer3_tablelist

This method is called at the end of the “*gap_primer3_create_primer_list*”-method, if primers were found. It creates the “Primer Table”-window and fills the tablelist with the created primer pairs. Depending on the setting of the interface options to create “Primer3_primer.passed” or “Primer3_no_primer.passed” files and the deletion of Primer3 input/output files, the accordant files will be created / deleted.

gap_primer3_update_info

It is linked to the tablelist object. When a row in the tablelist gets selected through a click, this method is invoked. It updates the information shown in the text field at the bottom of the “Primer Table”-window.

md5sum

It is executed through the method “gap_primer3_sqLiteDB” and creates an md5sum of a given list.

4.2.4 gap_multiplex.tcl

The Multiplex module creates lists of primer pairs, which can be used simultaneously in a multiplex-PCR experiment. As basis information serves a text file of primers, created through the Primer3 module, a primer table from a SQLite database or primer tags in the Gap4 database. Like in the Primer3 module the results can be written to a text file, as tags to the Gap4 database or as table to a SQLite database.

The multiplex primer candidates can be blasted against all contigs of the Gap4 database or a sequences file in FASTA format. By this means primers with multiple binding spots in the template DNA can be excluded. In every case all multiplex candidates are aligned against each other and get selected on basis of the alignment score. Additionally, the module provides the possibility to create a tree image in the PNG format, which shows the compatibility of the different primer pairs. This is only possible with a working installation of the R-Project.

The methods:

decr

This method is the opposite of the standard method "*incr*"; it decreases the value of an integer variable by 1.

gap_multiplex

This method is executed by invoking the module. It creates the interface of the Multiplex module.

gap_multiplex_alignPrim

This method is called by the methods "*gap_multiplex_ok*", "*gap_multiplex_create_mplx*" or "*gap_multiplex_treefile*". It gets the primer pair list and creates two new lists of primers; the first contains all primer sequences as they are in the primer pair list, the second list contains all this primer sequences in the reverse state. These two lists are given to "ntdpal", which returns a matrix with alignment scores of all primers. The results are returned to the method which called "*gap_multiplex_alignPrim*".

gap_multiplex_alignSeq

This method gets a list of all sequences in the Gap4 database and the primer pair list. It also creates a primer and a reverse primer list from the primer pair list. Then the two primer lists will be aligned against the Gap4 database sequences through “ntdpal” and a scoring matrix is returned.

This method is not used because of the time intensive alignment. Currently, BLAST is used for this kind of alignment. However, it is not as accurate as “ntdpal” but much faster.

gap_multiplex_blast

The method is invoked by the “*gap_multiplex_ok*”-method. It creates a BLAST database from a sequence list and a FASTA file with all primer sequences from the primer pair list. Subsequently, the “blastall” program is used to BLAST the created FASTA file against the created database. The BLAST results are parsed to a result list and returned to “*gap_multiplex_ok*”.

gap_multiplex_create_mplx

This method is executed through the “*gap_multiplex_tablelist_seq*”-method. It invokes the method “*gap_multiplex_alignPrim*” with a list of primer pairs which are selected in the tablelist in the “Blast Table”-window. The results are given to the “*gap_multiplex_selectPrim*”-method, which creates and returns the multiplex primer sets. At last the method “*gap_multiplex_tablelist*” is called to display this primer sets.

gap_multiplex_dir

This method allows browsing for and setting the Staden Package and R-Project paths in the Multiplex interface.

gap_multiplex_editEndCmd

It is linked to the tablelist object and is invoked by clicking on the “Use”-column of a row in the tablelist. It allows changing the state of the checkbox in this column.

gap_multiplex_export_primer

This method is executed through the “Export to File”-button in the “Multiplex Table”-window and writes all, through the “Use”-checkbox, selected primer pairs to a text file. The text file has the name set in the “Export to File”-box.

gap_multiplex_export_write

It is invoked through the "*gap_multiplex_export_primer*"-method, when the filename already exists. It opens a dialog which allows overwriting the existing file, to append the primer pairs to the existing file or to abort the writing progress.

gap_multiplex_get_primer_from_db

It is called by the "*gap_multiplex_ok*"-method and allows using the primer pairs marked in the Gap4 database for a multiplex primer search. It scans the whole database for all primer tags, set in the Multiplex interface, and writes the found pairs into the primer pair list. This list is returned to "*gap_multiplex_ok*".

gap_multiplex_invoke_editors

The method is linked to the tablelist object and invokes a contig-editor through a double click on a row in the tablelist. The contig-editor will show the location of the selected primer pair product.

gap_multiplex_mark_primer

By pressing the "Add to"-button, depending on the status of the "DB"- and "exp-File"-checkboxes, in the "Multiplex Table"-window, this method will add the selected primer pairs as tags to the Gap4 database or/and the experiment files. For marking the primers in the Gap4 database the right position in the right reading of the given contig has to be found.

gap_multiplex_ok

It is executed through the "OK"-button or the "Show R-Tree of Primer"-button of the Multiplex interface. Depending on the settings in the interface the method will create the primer pair list and sequence list and invokes the "Blast Table"- , the "Multiplex Table"-window or respectively opens the "Treefile.png".

For this the methods "*gap_multiplex_readPrimFile*", "*gap_multiplex_readSeqFile*", "*gap_multiplex_alignPrim*", "gap_multiplex_get_primer_from_db", "gap_multiplex_tree File", "gap_multiplex_blast", "gap_multiplex_alignPrim", "gap_multiplex_selectPrim" and/or "gap_multiplex_tablelist" are called.

gap_multiplex_populate_sequencelist

This method is linked to the primer tablelist in the “Blast Table”-window. When a row in the primer tablelist gets selected, the method clears the sequence tablelist and fills it with the primer specific blast results.

gap_multiplex_populate_tablelist

It will be executed through the buttons “Select All” or “Deselect All” in the tablelist. It empties the multiplex table and fills it anew with the same primer pairs, but with different state of the “Use”-column of the table.

gap_multiplex_readPrimFile

Reads the Primer3 output file, set in the Multiplex interface, and parses the contents to the primer pair list.

gap_multiplex_readSeqFile

Reads the sequence FASTA file, set in the Multiplex interface, and parses the contents to the sequence list.

gap_multiplex_safe

This method is processed after pressing the button “Save” in the Multiplex interface. It saves the parameter settings of the interface to a file in the “phobos_multiplex”-folder of the Staden Package. This file will be loaded automatically with every new start of the module.

gap_multiplex_selectPrim

The method will be executed at the end of the “*gap_multiplex_ok*”-method. It uses the primer pair list and primer score list to create a third list, which, for every single primer, contains the number of primers not conflicting with the current one. From this conflict list the primer with the highest value is taken, its value in the conflict list is set to 0 and all primer pair which conflict with this one are deleted from the primer pair and score lists. Now a new conflict list is created. This new list only contains primers which are useable with the already selected one. From this list the primer with the highest value is also taken and the cycle starts anew. This will happen until there are no primers left in the primer pair list.

That way, depending on the “Number of Multiplex Tables”-value in the Multiplex interface, a number of tables is created and returned to the “*gap_multiplex_ok*”-method.

gap_multiplex_sqLiteDB

This method is invoked through the “Export to SQLiteDB”-button in the “Multiplex Table”-window. It opens or creates a SQLite database with the name set in the “Export to SqLiteDB name”-box in the window. It will enter the selected rows as a new list in the SQLite database. To verify that the new table is not already in the database an md5sum of the list is created and compared to the lists already in the database.

gap_multiplex_tablelist

This method is called at the end of the “*gap_multiplex_ok*”-method, if multiplex primer sets were found. It creates the “Multiplex Table”-window and fills the tablelist with the compatible primer pairs.

gap_multiplex_tablelist_seq

It will be invoked by the “*gap_multiplex_ok*”-method, if the “Check against sequences”- and “Show Primertable for selection”-checkboxes are marked. It creates the “Blast Table”-window and fills a tablelist with all primers and through the method “*gap_multiplex_populate_sequencelist*” the other with the appropriate BLAST hits.

gap_multiplex_treeFile

This method is called when the button “Show R-Tree of Primer” in the Multiplex interface is clicked. It creates a text file with a score matrix of the selected primer pairs. If a version of the “R Project” is installed and the R-path, in the Multiplex interface, correctly set, a png image of a primer tree is created and opened. The executed R-script can be found in the “phobos_multiplex”-folder of the Staden Package.

gap_multiplex_update_info

It is linked to the tablelist object. When a row in the tablelist gets selected through a click, this method is invoked. It updates the information shown in the text field at the bottom of the “Multiplex Table”-window.

md5sum

It is executed through the method “gap_multiplex_sqLiteDB” and creates an md5sum of a given list.

4.2.5 gap_sqLiteDB.tcl

This module provides the interface for SQLite databases created with the modules of this project. It allows editing the databases by addition or deletion of tables and table entries. It also allows adding a comment to tables or chancing the ones already existing. Additionally, the tables of a database can be used as input for other modules of the project.

The methods:

gap_sqLiteDB

This method is executed by invocation of the module. It creates the interface to the SQLite module.

gap_sqLiteDB_create

It is called through the method "*gap_sqLiteDB_tablelist*", by clicking one of the buttons "New Table" or "Enter selected Rows" in a "SQLite Table"-window. Clicking the button "New Table" will create a new table of the same type as the active one and writes it to the database. This table contains no entries. With the button "Enter selected Rows" the rows, marked in the "Use"-column of the tablelist, can be added to another table with the id-number set in the text field behind the button.

gap_sqLiteDB_del

This method can be executed from the method "gap_sqLiteDB_ok". It allows deleting a selected table from the database.

gap_sqLiteDB_delete

It is called by the method "*gap_sqLiteDB_tablelist*" and allows to delete the whole table or to delete selected rows from the table.

gap_sqLiteDB_editEndCmd

It is linked to the tablelist object and is invoked by clicking on the "Use"-column of a row in the tablelist. It allows changing the state of the checkbox in this column.

gap_sqLiteDB_export_primer

This method is executed through the “Export to File”-button in a primer or multiplex table and writes all, through the “Use”-checkbox, selected primer pairs to a text file. The text file has the name set in the “Export to File”-box.

gap_sqLiteDB_export_repeats

This method is executed through the “Export to File”-button in a repeat table and writes all, through the “Use”-checkbox, selected repeats to a text file. The text file has the name set in the “Export to File”-box.

gap_sqLiteDB_export_write

It is invoked through the “*gap_sqLiteDB_export_primer*”-method, when the filename already exists. It opens a dialog which allows overwriting the existing file, to append the primer pairs to the existing file or to abort the writing progress.

gap_sqLiteDB_export_write_rept

It is invoked through the “*gap_sqLiteDB_export_repeats*”-method, when the filename already exists. It opens a dialog which allows overwriting the existing file, to append the repeats to the existing file or to abort the writing progress.

gap_sqLiteDB_get_list

Called by many different methods it extracts and returns all elements of the desired list from the SQLite database. To do this the method needs the id-number and the type of the table in question.

gap_sqLiteDB_invoke_editors

The method is linked to the tablelist object and invokes a contig-editor through a double click on a row in the tablelist. The contig-editor will show the location of the selected primer pair product.

gap_sqLiteDB_invoke_editors_rept

The method is linked to the tablelist object and invokes a contig-editor through a double click on a row in the tablelist. The contig-editor will show the location of the selected repeat.

gap_sqLiteDB_invoke_list_editors

The method is linked to the tablelist objects of the "*gap_sqLiteDB_ok*"-method and calls the method "*gap_sqLiteDB_tablelist*" through a double click on a row in a tablelist. The called method will display the selected table.

gap_sqLiteDB_mark_db_exp

By pressing the "Add to"-button, depending on the status of the "DB"- and "exp-File"-checkboxes, in the "Table"-window, this method will add the selected primer pairs / repeats as tags to the Gap4 database or/and the experiment files. For marking the primers / repeats in the Gap4 database the right position in the right reading of the given contig has to be found.

gap_sqLiteDB_multiplex

It is invoked through the "Create Multiplex"-button in a "Primer Table"-window of a SQLite database. The method creates a primer list of the selected primer pair in the tablelist and then invokes an instance of the Multiplex module with this primer list for multiplex creation.

gap_sqLiteDB_ok

This method is executing after opening a SQLite database. It creates the SQLite interface. It displays three tablelists, this lists contain the different kinds of tables in the database (multiplex, primer and repeat lists). Through a double click on a row of one tablelist the selected table will be displayed.

gap_sqLiteDB_populate_tablelist

It will be executed through the buttons "Select All" or "Deselect All" in the tablelist. It empties the table and fills it anew with the same primer pairs / repeats, but with different state of the "Use"-column of the table.

gap_sqLiteDB_primer3

It is invoked through the "Create Primer"-button in a "Repeat Table"-window. The method creates a target list of the selected repeat rows in the tablelist and then invokes an instance of the Primer3 module with this target list for primer creation.

gap_sqLiteDB_reload

This method, executed through the "Reload"-button in the SQLite interface, allows reloading the database. By working with the database it is possible to open the Primer3 or Multiplex module and add through those new tables to the database. These tables will not be shown in the SQLite interface until reloading the database.

gap_sqLiteDB_sqLiteDB

This method is invoked through the "Export to SQLiteDB"-button in a table window. It opens or creates a SQLite database with the name set in the "Export to SqLiteDB name"-box in the window. It will enter the selected rows as a new list in the SQLite database. To verify that the new table is not already in the database an md5sum of the list is created and compared to the lists already in the database.

gap_sqLiteDB_tablelist

This method creates a window which displays the contents a table. The appearance of the window depends on the kind of table which shall be displayed (multiplex, primer or repeat). Multiplex and primer tables have a text field at the bottom of the window, which shows some additional information the selected primer pair. Repeat tables have a second list in place of the text field. This list shows all primer pairs from multiplex or primer tables in the database, which enclose the selected repeat.

gap_sqLiteDB_tablelist_primer

This method is linked to the repeat tablelist and is executed when a row of the table gets selected. It clears the primer table of the window and fills it anew with primers from the database appropriate to the selected microsatellite.

gap_sqLiteDB_update_info

It is linked to the tablelist object. When a row in the tablelist gets selected through a click, this method is invoked. It updates the information shown in the text field at the bottom of the "Multiplex Table"-window.

md5sum

It is executed through the method "gap_sqLiteDB_sqLiteDB" and creates an md5sum of a given list.

4.2.6 gap_DBtoFASTA.tcl

This module was developed to allow the export of all contigs in a Gap4 database to a FASTA file. To do this there is only the “gap_DBtoFASTA” method. Through a standard method of Gap4 it creates a file with all consensus sequences of the database. This file is used to create a FASTA file including all contig sequences.

4.2.7 ntdpal

The C code *ntdpal* originates from the Primer3 sources [Rozen & Skaletsky 2000]. In Primer3 it is used to align two primer sequences against each other. It will align the forward against the reverse primer and both against their reverse complement. Through this ntdpal creates scores which are used in Primer3 to compute penalty values as quality marks for the primer pairs.

To apply ntdpal for the selection of primer pairs useable in a multiplex-PCR it had to be rewritten. Formerly, ntdpal allowed the input of two primer sequences. The ntdpal used by the Multiplex module gets two primer sequence lists as input. The second list contains the reverse complements of all primer sequences in the first list. It returns not a single score but a score matrix, which is used by the Multiplex module to select primer pair combinations for usage in a multiplex-PCR.

4.2.8 R-Project

In this project R is used to create a tree from a text file of primer pairs. This tree shows a relative usability of the primer pairs in a multiplex-PCR. Primer pairs on different branches of the tree are more likely to interact with each other than pairs on the same branch. Normally this means primer pairs from the same branch could be used in a multiplex-PCR. There can be some exceptions. To build-up the tree average scores for every primer pair are created, through this it can happen, that two primer pairs, which will interact with each other, appear in the same branch of the tree. As example: the forward primer works perfectly with both primers of the other pair, but the reverse primer resembles the binding site of the forward primer of the other pair. Through the average primer pair scores the two pairs would seem usable with each other and would appear relatively near each other in the tree.

The R script used for the tree creation can be found in the “phobos_multiplex”-folder of the Staden Package.

5 Laboratory tests

5.1 Materials and methods

5.1.1 *Phaeocystis*

For the multiplex-PCR experiments DNA from *Phaeocystis antarctica* was used.

The alga *Phaeocystis* is distributed worldwide and is an important part of the phytoplankton, but is also seen as a nuisance [Baumann *et al*. 1994; Schoemann *et al*, 2005; Veldhuis and Wassmann, 2005].

Phaeocystis can appear as single flagellated cells [Whipple *et al*, 2005] or in huge colonies sometimes of more than 3 cm in diameter [Shen *et al*, 2004].

It is a bloom forming alga which can pollute large areas after its bloom crashing. When the bloom decays, the remains start to foam and are carried to the shore [Lancelot *et al*, 1987]. *Phaeocystis* accounts for a great part of the global sulphur budget through the release of dimethylsulfide propionate (DMSP) [Baumann *et al*, 1993].

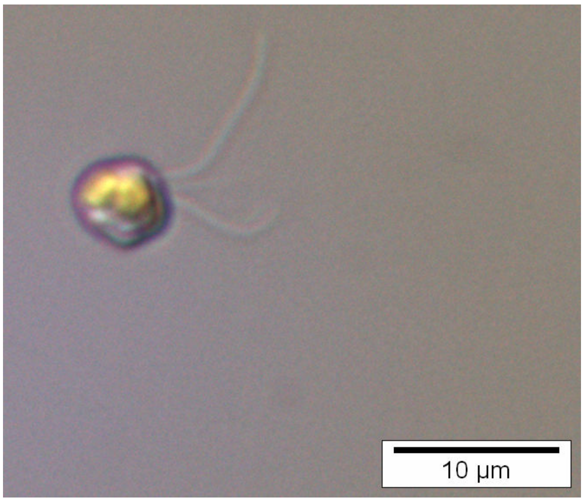


Fig. 5.1.1.a: Phaeocystis antarctica*, flagellate cell; picture by Steffi Gäbler-Schwarz*

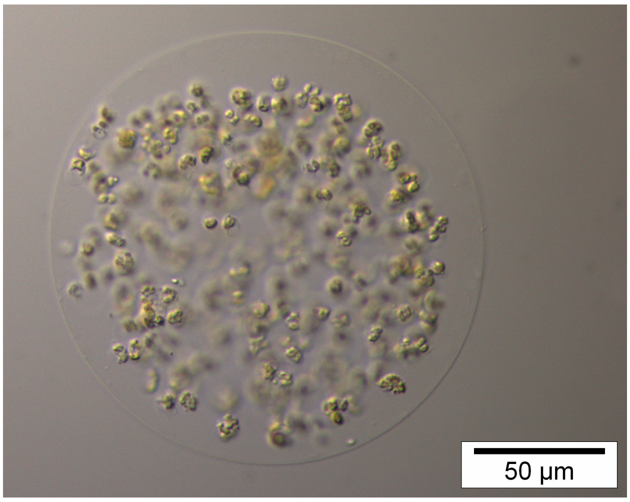


Fig. 5.1.1.b: Phaeocystis antarctica*, cell colony; picture by Steffi Gäbler-Schwarz*

5.1.2 PCR

The polymerase-chain-reaction, or PCR [Mullis, 1983], is a method for in vitro amplification of DNA. Only relative short DNA-segments are copied, up to 3000 bases (3kbp) in length.

The PCR is used in a number of areas, for example for the „Genetic Fingerprint“, gene cloning, mutagenesis or for identification of evolutionary relationships between organisms. The latter applies here in form of microsatellites.

To accomplish a PCR the following components are needed:

- the template DNA, with the segment for amplification
- two primers (short DNA sequences, approx. 20 bases long), which bind to the two single strands of the template DNA and, in this manner, confine the area for amplification
- DNA-polymerase, which binds to the primers and constructs the missing DNA-strand
- desoxynucleosidetriphosphate (dNTPs), the components for the synthesis of DNA-strands through the polymerase
- Mg^{2+}-ions, as co-factor for the polymerase
- buffer solutions, for an appropriate chemical environment for the reaction.

Figure 5.1.2.a shows a cycle in a PCR, the different steps are marked with red numbers.

1. Denaturation (melting): The double-stranded DNA is unravelled in two single-strands, happening through heating to 94-96°C. This will split the hydrogen bonds which hold the strands together. In the first cycle the DNA will be heated for a longer period to assure that only single-stranded DNA exists.
2. Primer hybridisation (annealing): After separating the strands the temperature is reduced, so the primer can attach to the DNA-strands. The temperature at this point depends on the used primers. If the temperature is too high the primers will not attach to the DNA, if it is to low they will attach to wrong places in the DNA-strand.
3. Elongation (extension): The DNA-polymerase fills the missing strands with free nucleotides. It starts at the 3'-ending of the attached primer and follows the DNA-strand. The primer will not detach, it makes up the beginning of the strand. The temperature depends on the used DNA-polymerase (between 68 °C and 72 °C). The time for this step is dependent on the used polymerase as well as the length of the DNA-fragment for amplification.
4. The cycle starts anew.

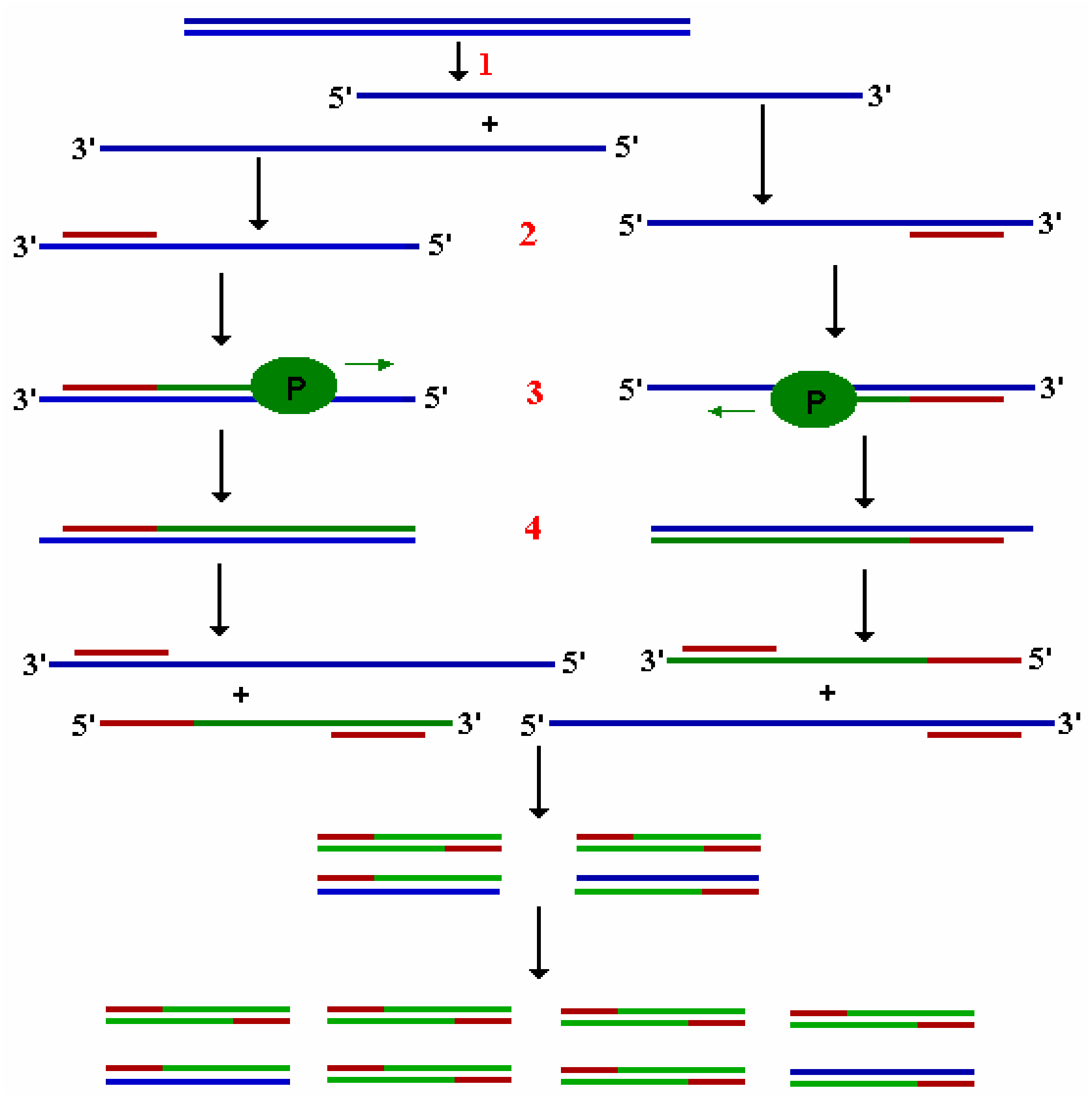


Fig. 5.1.2.a: PCR-cycle

5.1.3 Multiplex PCR

A multiplex-PCR works in the same way as a normal PCR. The difference is that more than one primer pair is used in the reaction. The usage of several primer pairs in one tube permits saving time and effort. There are numerous applications for multiplex PCRs from *in vitro* cloning to disease diagnosis [Elnifro *et al*, 2000].

It is important that the different primers used in a multiplex-PCR cannot attach to each other. This would reduce the efficiency of the reaction and probably inhibit it altogether. Also it is important that the primers do not have multiple binding spots in the template-DNA, this would reduce the primer efficiency and possibly allow the amplification of unwanted product-sequences. In addition, the PCR mixture and conditions (annealing temperature and cycle number) are very important, they require individual adjustment for decent multiplex results [Henegariu *et al*, 1997]

Normally the primers in a multiplex-PCR have to be marked in some way (see section 5.1.4); otherwise the PCR-products could not be distinguished from one another, except the products differ in size more than a few base pairs.

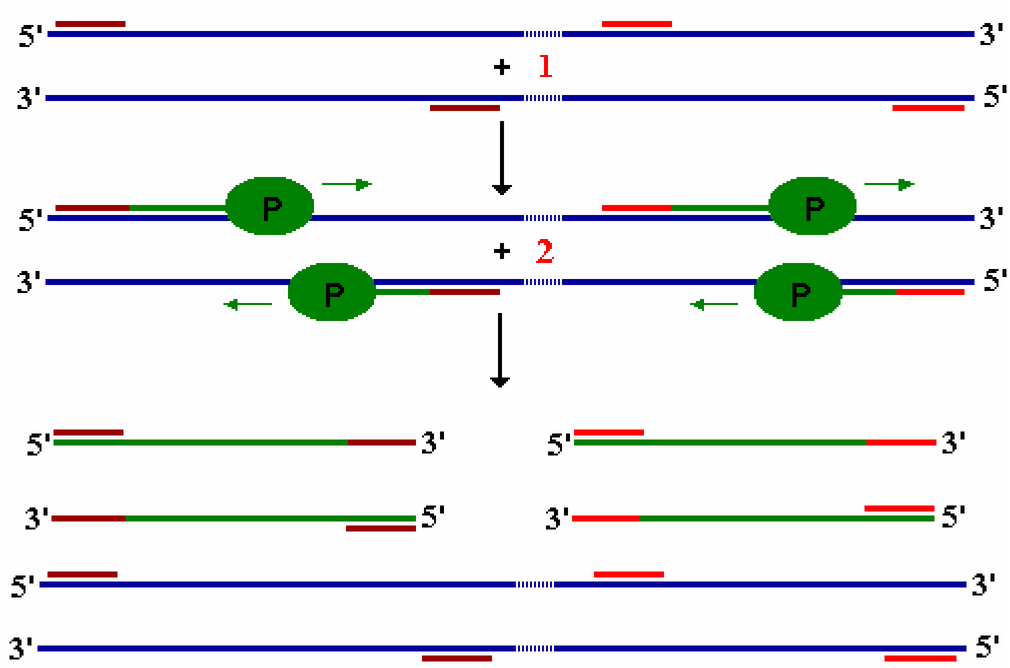


Fig. 5.1.3.a: Multiplex-PCR with two primer pairs

1.) Primers anneal to the DNA single strands. 2.) DNA-polymerase extends the primers.

3.) In the next cycle (after denaturation) four product strands and the two template strands are available for primerhybridisation.

5.1.4 Dye labelled primer

Dye labelled primers contain a fluorescence group at their 5’-end. These groups - called dyes - are of considerable size and influence the binding ability of the primer. Fluorescein, NBD, Tetramethylrhodamin, Texas Red and others can be used as dyes. They are coupled to a nucleotide and are accepted by the DNA-polymerase as 2’-desoxynucleotide or as 2’3’-didesoxynucleotide. The different dyes can be detected through their emissions after excitation by laser light.

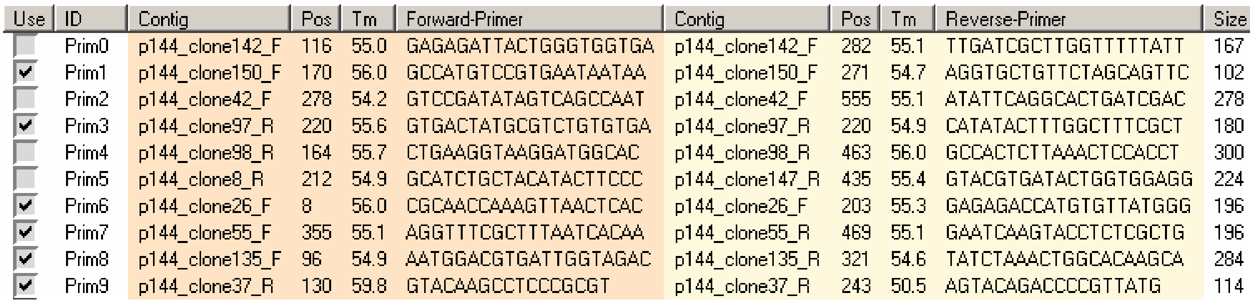
5.2 Test runs

There were already 23 primer pairs available (Eurofins MWG Operon, *http://www.eurofinsdna.com*), which were designed with earlier versions of the Phobos and Primer3 modules for Gap4 (unpublished data by Steffi Gäbler-Schwarz). Some of these primers were modified and differed from the original results of the Primer3 module.

17 of the 23 primer pairs could be found again, identical to the available ones. For this the standard parameters of the Primer3 module were sufficient.

Those 17 were tested for usability in a multiplex-PCR. 7 primer pairs showed multiple binding spots in their Gap4 database. The other 10 provided several lists of possible combinations up to 7 pairs in one multiplex-PCR experiment. By decreasing the score value, from the standard of 8 down to 3, primer combinations were listed which should show minimal interaction of the different primers during a PCR. A score value under 3 delivered no possible combinations at all.

6 of the 10 primer pairs possible for a multiplex were tested in single PCR experiment (Fig. 5.2.a shows the specifics of all 10 primer pairs and marks out the tested ones). After several tests and modifications of the concentrations of dNTPs and primers and changes of the annealing temperature the PCR results were satisfactory. The primer pairs most appropriate for a multiplex-PCR (Prim7, Prim8 and Prim9) did not deliver the best results in single PCR.



Use	ID	Contig	Pos	Tm	Forward-Primer	Contig	Pos	Tm	Reverse-Primer	Size
□	Prim0	p144_clone142_F	116	55.0	GAGAGATTACTGGGTGGTGA	p144_clone142_F	282	55.1	TTGATCGCTTGGTTTTTATT	167
☑	Prim1	p144_clone150_F	170	56.0	GCCATGTCCGTGAATAATAA	p144_clone150_F	271	54.7	AGGTGCTGTTCTAGCAGTTC	102
□	Prim2	p144_clone42_F	278	54.2	GTCCGATATAGTCAGCCAAT	p144_clone42_F	555	55.1	ATATTCAGGCACTGATCGAC	278
☑	Prim3	p144_clone97_R	220	55.6	GTGACTATGCGTCTGTGTGA	p144_clone97_R	220	54.9	CATATACTTTGGCTTTCGCT	180
□	Prim4	p144_clone98_R	164	55.7	CTGAAGGTAAGGATGGCAC	p144_clone98_R	463	56.0	GCCACTCTTAAACTCCACCT	300
□	Prim5	p144_clone8_R	212	54.9	GCATCTGCTACATACTTCCC	p144_clone147_R	435	55.4	GTACGTGATACTGGTGGAGG	224
☑	Prim6	p144_clone26_F	8	56.0	CGCAACCAAAGTTAACTCAC	p144_clone26_F	203	55.3	GAGAGACCATGTGTTATGGG	196
☑	Prim7	p144_clone55_F	355	55.1	AGGTTTCGCTTTAATCACAA	p144_clone55_R	469	55.1	GAATCAAGTACCTCTCGCTG	196
☑	Prim8	p144_clone135_F	96	54.9	AATGGACGTGATTGGTAGAC	p144_clone135_R	321	54.6	TATCTAAACTGGCACAAGCA	284
☑	Prim9	p144_clone37_R	130	59.8	GTACAAGCCTCCCGCGT	p144_clone37_R	243	50.5	AGTACAGACCCCGTTATG	114

Fig. 5.2.a: List of primer pairs with single blast hits. The primer pairs marked in the "Use"-column were tested.

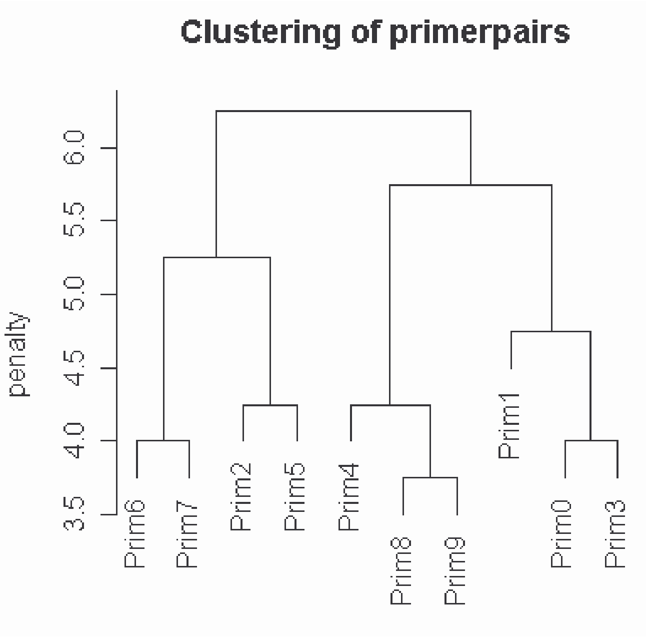


Fig. 5.2.b: Tree created with an R-script.

Only two primer pairs worked well enough under the same conditions to be used in a multiplex experiment (Prim6 and Prim9). In order to get acceptable multiplex results with the other primer pairs many more adjustment tests will be necessary.

The specifics of the used primer pairs for the multiplex-PCR, the PCR program and mix are shown in the tables 5.2.a to 5.2.c.

Fig. 5.2.b shows a clustering of the primer pairs depending of how well they would work together in a multiplex. The tree serves only as guideline because it uses average values for each primer pair against the others (see section 4.2.7).

Tab. 5.2.a: Multiplex tested primer pairs

		Primer	**Length**	**Melting Tm**	**GC**	**Product size**
Clone26_F	Forward	CGCAACCAAAGTTAACTCAC	20	56,0	45,0	196
	Reverse	GAGAGACCATGTGTTATGGG	20	55,3	50,0	
Clone37_F	Forward	GTACAAGCCTCCCGCGT	17	59,8	64,7	114
	Reverse	AGTACAGACCCCGTTATG	18	50,5	50,0	

Tab. 5.2.b: PCR-Mix;
(Eppendorf, **Sigma Aldrich;***NEB)*

Component	µl
PCR-Buffer *	2
dNTPs (1mM)*	2
Betaine (5M)**	2
BSA (1:10)***	1
Forward Primer 1	1
Reverse Primer 1	1
Forward Primer 2	1
Reverse Primer 2	1
H_2O	7,7
Taq-Polymerase* (5 U/µl)	0,3
DNA-Template	1
Total:	**20µl**

Tab. 5.2.c: Used PCR program

Steps	MS54C50	
1	94°C	1 min
2	94°C	15 sec
3	54°C	20 sec
4	72°C	30 sec
5	jump to 2	50 cycles
6	72°C	10 min
7	4°C	On hold

The two primer pairs in the multiplex were also used in a fragment analysis. Both forward primers were labelled with a fluorescent dye at the 5'-end by Thermo Electron Corp. (*http://www.thermo.com*). Clone26_F was HEX labelled and Clone37_F was 6-FAM labelled.
The single PCRs were made under the same conditions as the ones with the unlabelled primers. For the multiplex-PCR experiments higher annealing temperatures (57°C and 60°C) were used.
For the fragment analysis 1 µl of the every PCR product was pipetted in an ABI plate containing 15µl Hi-Di Formamide (Applied Biosystems, *http://www.appliedbiosystems.com*) and 0,3 µl GeneScan-500 ROX Size Standard (Applied Biosystems). The mixture was denatured for 3 minutes by 95°C.
Fragment analysis was done by electrophoreses using a capillary sequencer (ABI PRISM 3100 AVANT, Applied Biosystems). Data visualisation was done using Genemapper V4.0 (Applied Biosystems).

5.3 Results

5.3.1 Unlabelled primers

For visualization of the unlabelled PCR products 3.5% Metaphor-agarose gels (BMA; *www.bmaproducts.com*) were used. The gels were stained with Ethidium bromide (EtBr). To visualize the right fragment size a 100bp DNA-ladder (peqGold Plus, Peqlab; *http://www.peqlab.de*) was used on the gels. It consists of 14 fragments from 100bp to 3000bp.

The picture 5.3.1.a shows the PCR results of the Clone26_F primer pair. This primer pair worked at every tested annealing temperature and showed the same results only with variations in their intensity. The expected product size was 196bp. With all DNA samples a product of this size was obtained. The bands are marked red in the picture. Samples A1-3, DE2 and 22_12 showed some additional products. These products may refer to heterozygous samples, but could also be results of unspecific bindings. The bands at the bottom of the gel are primer-dimer clouds.

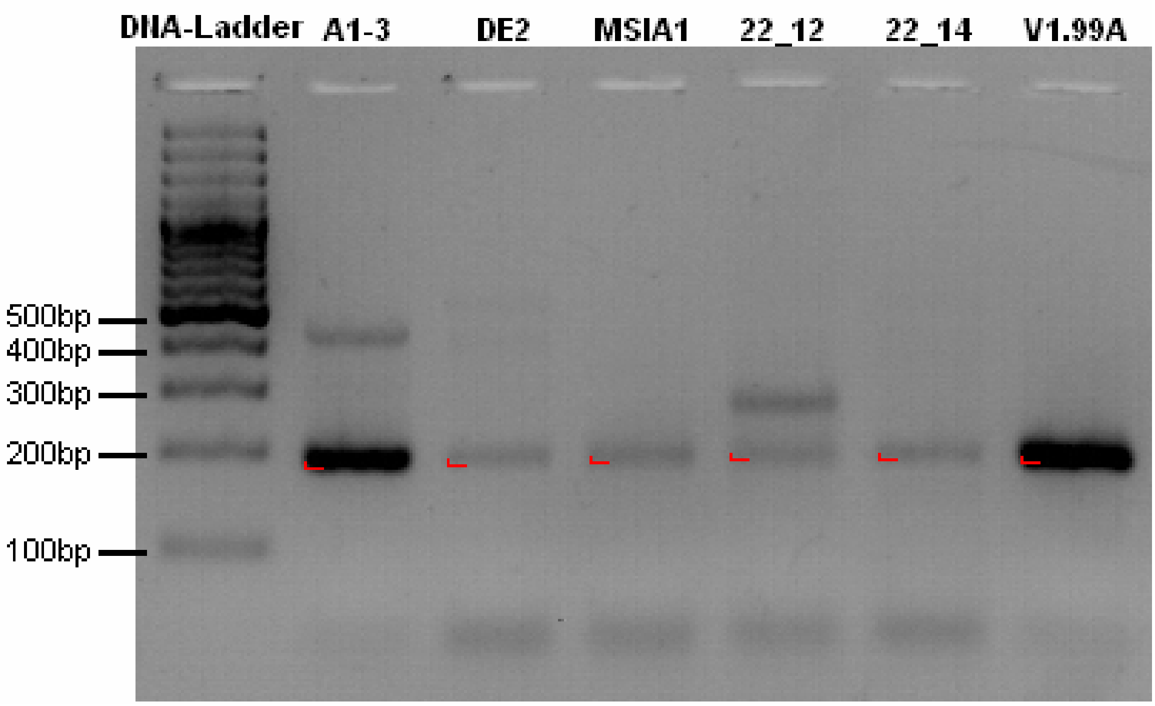


Fig. 5.3.1.a: Single PCR with Clone26_F, expected product size: 196bp. The red tagged bands represent the anticipated products.

Picture 5.3.1.b shows the PCR results of the Clone37_F primer pair. In contrast to the other tested primer pairs it worked also at higher temperatures. The expected product should have a size of 114bp, here only three of the six samples showed a product near the expected height. The expected products are marked yellow in the picture. The product in A1-3 appears to be much higher than anticipated, but is likely to be the right product with an insert. The same goes for the marked product in MSIA1. V1.99A shows no product at all in between 100bp and 500bp. The other products above 500bp might be results of unspecific bindings. The bands below 100bp are primer-dimer clouds.

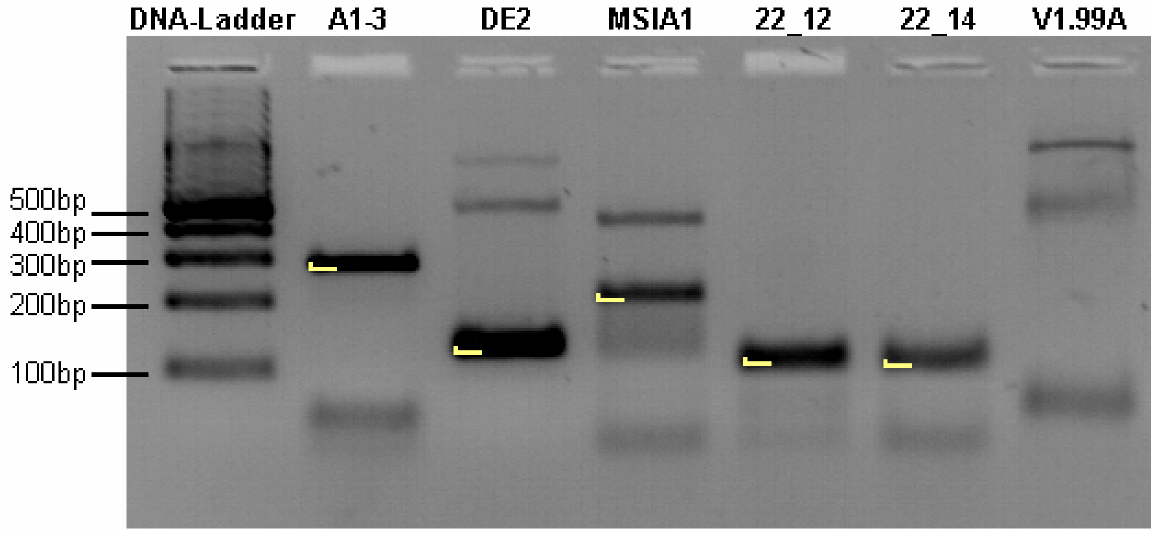


Fig. 5.3.1.b: Single PCR with Clone37_F, expected product size: 114bp. The yellow tagged bands represent the anticipated products.

Picture 5.3.1.c shows the multiplex-PCR results of the Clone26_F and Clone37_F primer pairs. The expected products of the Clone26_F primer pair should have a size of 196bp and are marked red in the picture, the expected products of the Clone37_F primer pair are marked yellow and should have a size of 114bp. All bands of the anticipated product of Clone26_F are also visible on the gel of this multiplex-PCR. The bands of Clone37_F in the samples DE2, 22_12 and 22_14 are also visible. In MSIA1 one product band can be seen which could contain the products of both primer pairs. In A1-3 a really weak band can be seen a little under 300bp, this might be the product of the Clone37_F primer pair. In DE2, 22_14 and V1.99A bands are visible which do not appear in the gels of the single PCR, but also many bands of the bands in Fig. 5.3.b do not reappear in this multiplex-PCR.

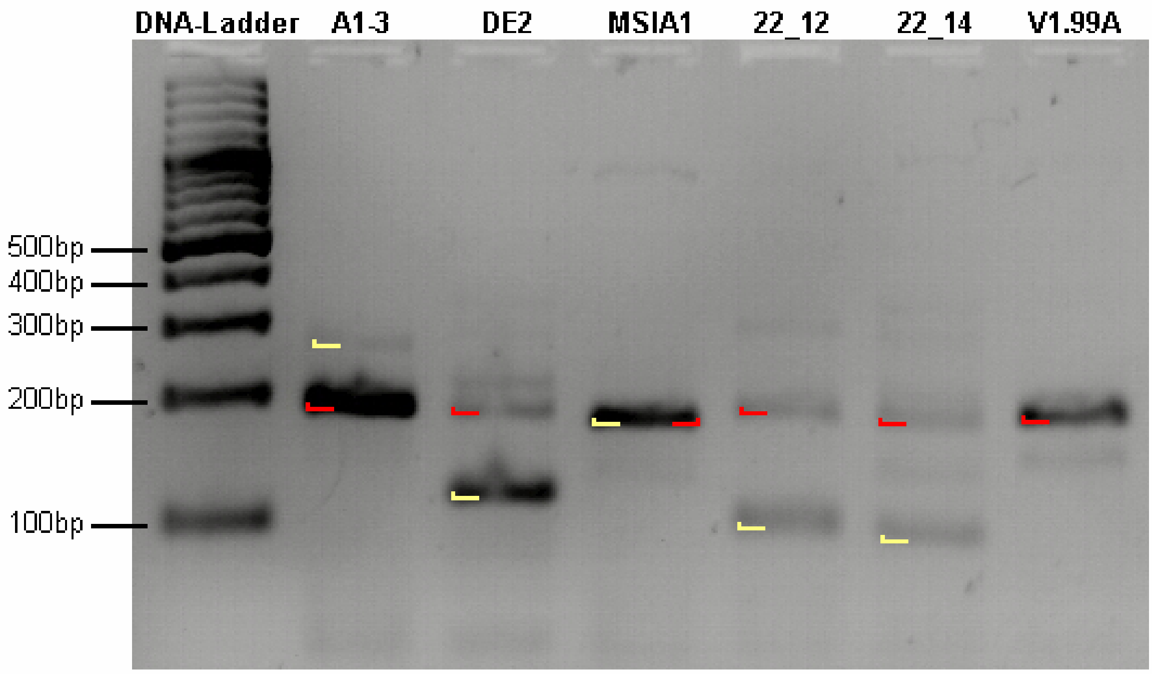


Fig. 5.3.1.c: Multiplex-PCR with Clone26_F and Clone37_F. The yellow and red tagged bands represent the anticipated products.

5.3.2 Labelled primers

Figure 5.3.2.a shows the results of the single and multiplex PCRs with the two labelled primers pairs for DNA sample 22_12. The PCR results of Clone26_F are displayed in green and the results of Clone37_F in blue. The pictures show the expected peaks and also additional ones. The multiplexes include peaks which contain both labelled primers.

The two peaks of the single PCR of Clone26_F between 250bp and 275bp do not appear in the two multiplex experiments. The expected peaks in the multiplex intensify with the rise of temperature and the intensity of the peaks containing both labelled primers decrease. A further rise of the temperature could prevent the wrong pairing completely.

Figure 5.3.2.b shows the results of the single and multiplex PCRs for DNA sample A1-3. The pictures show always the expected peak for Clone26_F. The single PCR of Clone37_F shows several distinct peaks, including one at 275 bp which can be the one seen on the gel (see Fig. 5.3.1.b). The multiplex PCR with an annealing temperature of 57°C shows only the two expected peaks. In the multiplex at 60°C the peak for Clone37_F disappears which indicates a temperature limit for sample A1-3 much lower than the limit for sample 22_12.

The other DNA samples also include peaks which contain both labelled primers. Those peaks are reduced with the rise of temperature.

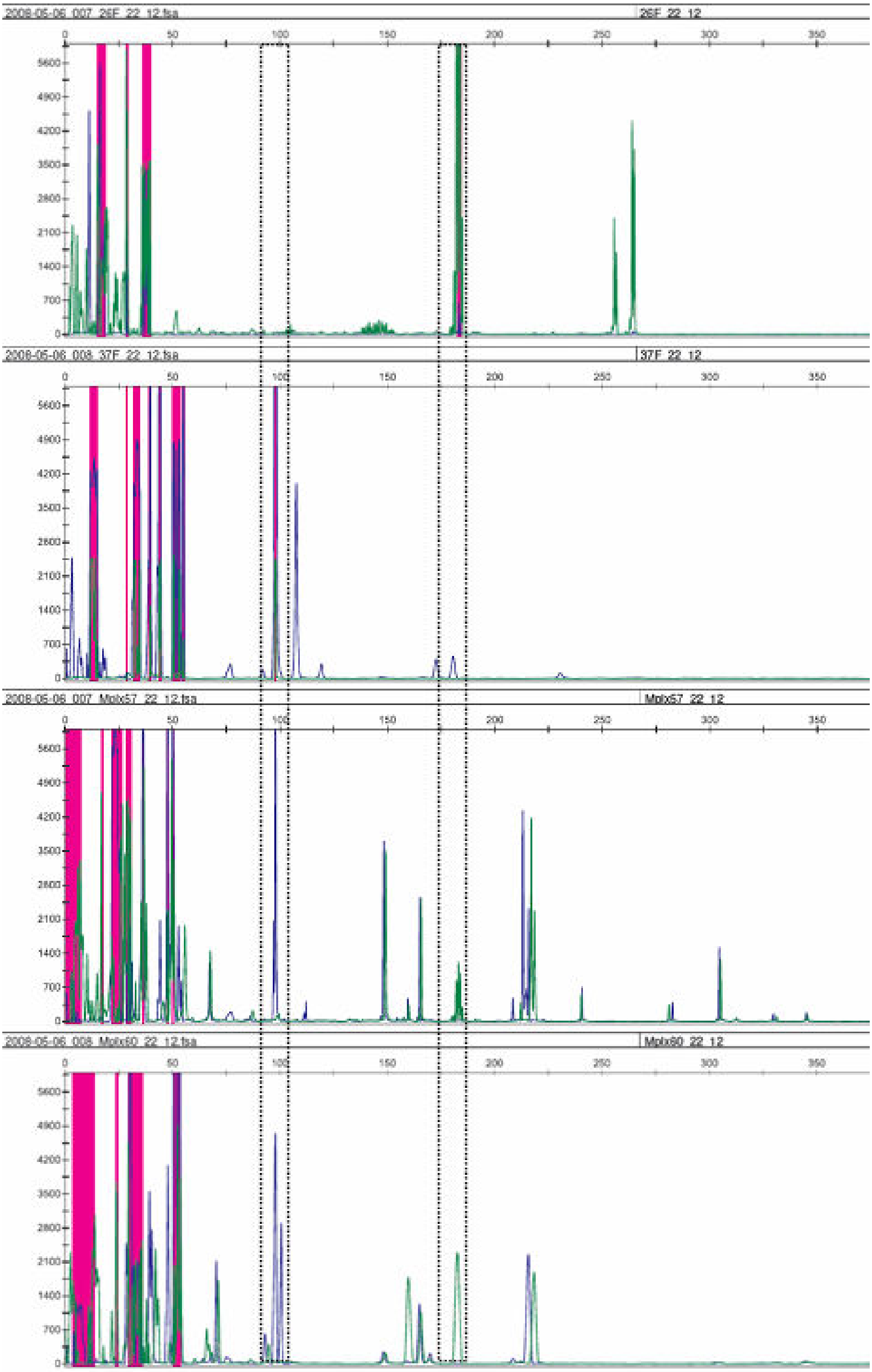


Fig. 5.3.2.a: Results of single and multiplex-PCRs for DNA sample 22_12. Order: Clone26_F (green), Clone37_F (blue), multiplex at 57°C for annealing and multiplex at 60°C.

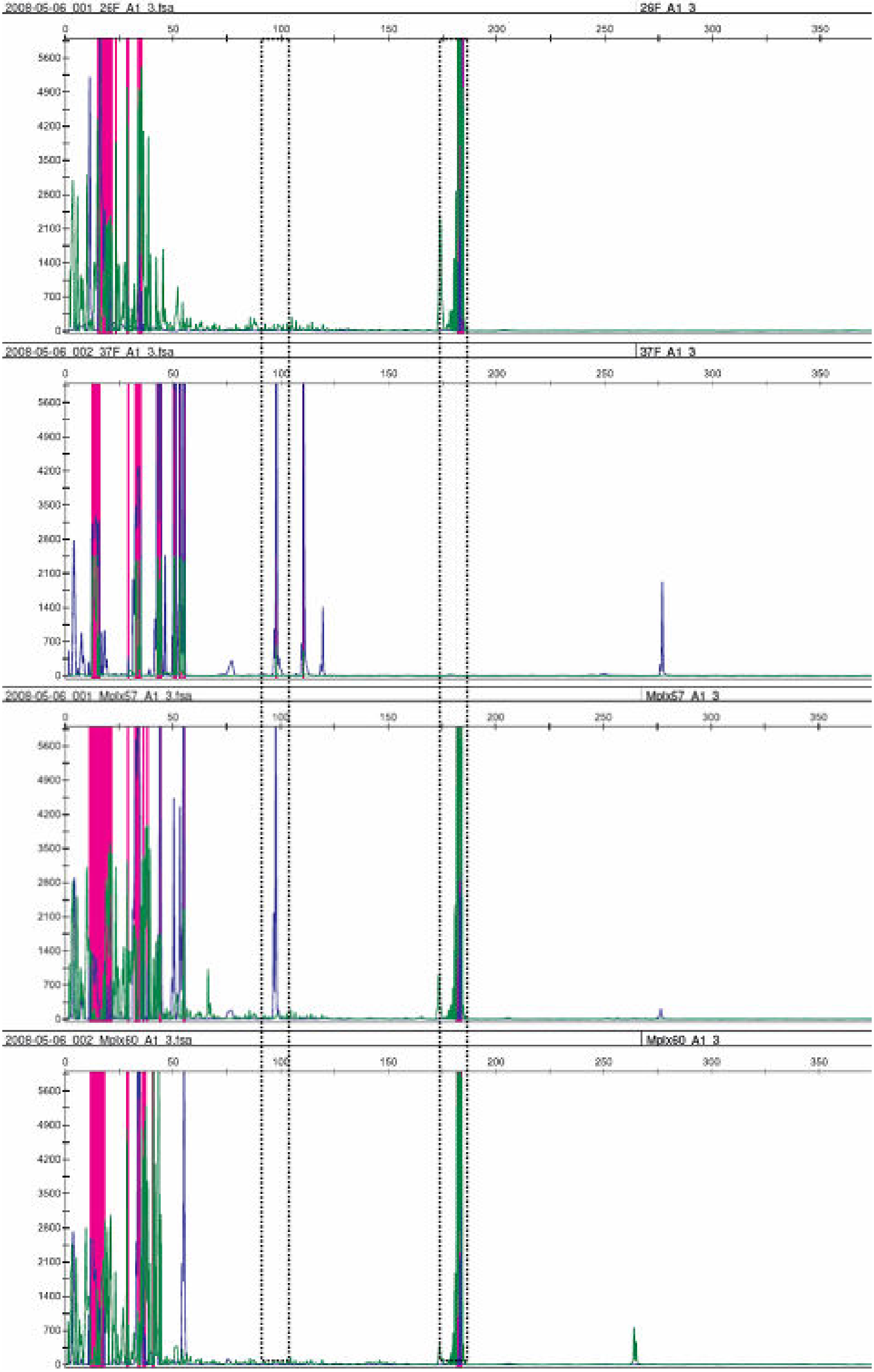


Fig. 5.3.2.b: Results of single and multiplex-PCRs for DNA sample A1-3. Order: Clone26_F (green), Clone37_F (blue), multiplex at 57°C for annealing and multiplex at 60°C.

5.4 Further tests of the system

The Gap4 modules for microsatellite search and primer design were already used and tested by Florian Leese [Leese *et al*, 2008] and Steffi Gäbler-Schwarz. The results of Gäbler-Schwarz are currently unpublished.

Leese created a short-insert genomic library of *Serolis paradoxa*. 167 inserts of clones from this library were sequenced and assembled with the normal shotgun assembly algorithm of Gap4, allowing 10% of mismatches in the flanking regions. 40 sequences were redundant, that left 127 unique inserts, which were searched for microsatellites with a perfection of 95% or higher. For 22 appropriate loci primer pairs were developed automatically. 21 of these generated distinct PCR products, which is comparably high. 15 loci could be reliable genotyped (68%), 13 of these were polymorphic.

6 Discussion

The modular extensions for the Staden package invented in this project provide a fast and effective workflow for interactive microsatellite primer design, even at high data throughput. The project shows a number of improvements compared to other available tools for similar purposes. It is embedded in an interactive software environment providing functionality concerning basic analysis steps (e.g. base calling, vector and quality clipping, assembly with redundancy checking) and user interaction (e.g. contig editor, manual changes to assembly). The project offers a full integration of repeat detection, flanking primer design and multiple primer design. These functions were not only used successively, but also independently and repeatedly with different specifications. Besides this, the project improved the data storage mechanism used by the Staden package through usage of SQLite, which now provides a much higher transparency of the created data.

Furthermore the integration of Phobos - being the most flexible and consistent repeat detection tool currently available - assures the quality of the processed microsatellites. Its abilities to find not only perfect but also imperfect SSRs and identify alternative repeat patterns at the same location are outstanding features which most other microsatellite detection tools perform poorly or not at all. To find imperfect SSRs is of special interest, because they are more common than perfect ones and are highly useful in population genetics.

The Troll module for Pregap4 [Martins *et al*, 2006] for example lacks the ability to find imperfect repeats completely and programs like Msatfinder [Thurston & Field, 2005] show a limited accuracy. Other tool for microsatellite marker design (msatcommander [Faircloth, 2007], read2marker [Fukuoka *et al*, 2005], SSR primer [Robinson *et al*, 2004]) showed also one or more of the following weaknesses: lack of redundancy checks, weak or inconsistent solutions for the repeat detection problem and lack of integration and interactivity.

For preventing redundancy in sequences of genomic libraries it is necessary to assemble these sequences. This happens here through the Staden package, but for a decent assembly the repeats, including the imperfect ones, have to be masked, otherwise the assembly will create a lot of wrong overlaps. To allow such a masking in the Gap4 assembly the Pregap4 Phobos module was invented, which tags the microsatellites in the single sequences. A special advantage of the Phobos module for Gap4 itself is, that it allows to create several lists of found microsatellites depending on the chosen characteristics (motif or repeat length, perfection). The repeats can be saved by the user individually and subsequently be used for primer design.

Primer3 is a very good tool for this purpose, but as a stand-alone version the user would have no possibility to validate the created primers. It would not be possible to interactively take a look at the contig the primer pair lies in or the microsatellite they enclose. The Primer3 module for Gap4 provides this possibility. It also enables the user to perform multiple rounds of primer design to successively add primer candidates to a list of the most promising ones.

The Multiplex module is the last step in the workflow. It relies on the primer comparison algorithm of Primer3 and also makes sure that there are no alternative binding spots for the tested primers in the available sequence data. This Blast option of the module can also be used for single primer design in order to reduce the occurrence of unwanted, additional PCR products. Like the other modules it can be used fast and easily multiple times on the same set of input data with varying parameters, thus allowing for an iterative refinement. This will result in several suggestions of possible multiplex primer combinations. However, how reliable the results of the module are has to be tested more extensively.

The last addition to this project, the SQLite module, allows the storage of all produced data in a SQLite database. The storage in such a database is very easy and data can be displayed again clearly arranged. With the Gap4 database this scale of data storage and the magnitude of accessibility would not be possible. The application of SQLite, with already integrated Tcl bindings, made it also possible to easily connect information of the Gap4 database with already stored data. Furthermore, it allows primer design and multiplex primer selection regarding the sequence data of a Gap4 database, without the necessity of the appearance of the used input data (repeats or primers) as tags in the Gap4 database.

The tests of Leese and Gäbler-Schwarz proof the effectiveness of the microsatellite primer design with the Staden package and the modules for Phobos and Primer3. The improved versions of these modules and the additional ones (Multiplex and SQLite) are further making the whole design process a lot easier and more transparent. Through the data storage with SQLite and the resulting table-handling the selection of appropriate loci for primer design is less time intensive as the direct search in a Gap4 database. Furthermore, the Multiplex module will make the work in the laboratory less time consuming and arduous. The biggest problem for the accuracy of the suggested primer combinations is that the whole genome of the examined organism is normally not available. The unavailable regions could contain binding sites for primers in the multiplex mixture and maybe result in unexpected PCR products of primers of different pairs.

The tests made in this project to proof the functionality of the Multiplex module provided not the results wished for, but they were adequate. The primers already available were not designed with a multiplex-PCR in mind and the project allowed a rather limited set of adjustment experiment. The adjustment of a multiplex-PCR consists of many steps and is a time consuming task [Henegariu *et al*, 1997].

One working multiplex-PCR mixture including two pairs of primers was demonstrated to work. The tests with the labelled primer pairs show that changes of the annealing temperature for a PCR cause profoundly different results. The to labelled forward primers amplified unwanted fragments, this only could be prevented with the knowledge of the whole genome sequence. But these products were reduced with the rise of the annealing temperature. With further experiments for the setting of the annealing temperature, PCR cycle number and especially the primer mix ratio, multiplex-PCR combinations of several of the tested primer pairs will be possible.

Additionally to further experiments to proof the functionality of the Multiplex module, a lot of minor alterations should be made within most of the software modules. Those changes would further improve the comfort of the software environment. The R script for the compatibility tree should be changed completely in order to resemble the compatibility of single primers and not the average one of primer pairs. A further refinement of the selection algorithm of the Multiplex module could further improve the quality of the results. At the moment only the primer interaction is taken into account. The other attributes (melting temperature of primer and product and GC-content) also have to be considered. This could help to ensure the functionality of the applied primer pairs under the same PCR conditions and would reduce the number of adjustment experiments. For now, this specification has to be done through the settings of the primer design module or a manual selection of the primer pairs for the input of the Multiplex module.

Alongside these a better algorithm for the sequence assembly than the normal shotgun assembly of Gap4 should be integrated. For this task, Phrap [Green & Ewing, 1994] seems to be a good choice because it is already available as a module in Gap4. The great disadvantage of Phrap is the absence of a function to mask any sequence region like repeats or vector parts. If such a function was integrated, the assembly in Gap4 would be considerably improved. A better assembly would also result in a better outcome of the Blast function of the Multiplex module.

It can be summarized that this project achieved its goals. It is written in the English language to be available on an international level and under the GPL (GNU General Public License). With Phobos for MS detection and the integration into the Staden package, with such extensive analysis features, it seems to be one of the best integrative bioinformatics tools for the design of microsatellite markers available at the moment. The function for multiplex-PCR primer selection is hardly available in any other tool, especially not in such a simple and comfortable way. On the other hand, the project still has potential for improvements; an upgraded algorithm for multiplex primer selection and a superior one for the assembly are only the most obvious. Furthermore, an automated dot-plot function (to ensure the singularity of primer binding spots) and online access to sequence databases (access to whole genomes) could be of use. The database function could be expanded to allow the storage of results for already tested primer pairs. Such results could be the optimal PCR mixture and program or gel pictures. Information like this could be used as basis for the adjustment of untested primer pairs for single or multiplex PCR experiments.

Literature

Altschul SF, Gish W, Miller W, Myers E and Lipman DJ (1990) Basic Local Alignment Search Tool. *Journal of molecular biology* London: Academic Press., **215**, 403-410

Baumann MEM, Brandini FP and Staubes R (1993) The influence of light and temperature on carbon specific DMS-release by cultures of *Phaeocystis antarctica* and three Antarctic diatoms. *Mar. Chem.* **45**, 56–78

Baumann MEM, Lancelot C, Brandini FP, Sakshang E and John DM (1994) The taxonomic identity of the cosmopolitan prymnesiophyte phaeocystis: a morphological and ecophysiological approach. *J. Mar. Sys.* **5**, 5–22.

Elnifro EM, Ashshi AM, Cooper RJ, Klapper PE (2000) Multiplex PCR: optimization and application in diagnostic virology. *Clin. Microbiol. Rev.* **13**, 559– 570

Ewing B and Green P (1998) Base-calling of automated sequencer traces using phred. I. Accuracy Assessment. *Genome Res.* Vol. 8, Issue **3**, 175-185, March 1998.

Ewing B and Green P (1998) Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Res.* Vol. 8, Issue **3**, 186-194, March 1998.

Faircloth BC (2007) Msatcommander: detection of microsatellite repeat arrays and automated, locus-specific primer design. *Molecular Ecology Notes* (OnlineEarly Articles). doi:10.1111/j.1471-8286.2007.01884.x

Fukuoka H, Nunome T, Minamiyama Y, Kono I, Namiki N, and Kojima A (2005) read2Marker: a data processing tool for microsatellite marker development from a large data set. *BioTechniques,* **39**, 472-476

Gotoh O (1982) An improved algorithm for matching biological sequences. *Journal of molecular biology* London: Academic Press., **162**, 705-708

Green P and Ewing B (1994) PHRAP documentation. [http://www.phrap.org].

Henegariu O, Heerema NA, Dlouhy SR, Vance GH and Vogt PH (1997) Multiplex PCR: critical parameters and step-by-step protocol. *BioTechniques*, **23,** 504-511

Hipp R "The Tcl interface to the SQLite library." [http://www.sqlite.org/tclsqlite.html].

Jarne P and Lagoda JL (1996) Microsatellites, from molecules to populations and back. *Trends Ecol. Evol.***11**, 424–429

Kraemer, L (2007) Integration of programs for the identification of microsatellites into the Staden Package and integrated workflows for primer design of genetic markers. Unpublished project work.

Lancelot C, Billen G, Sournia A, Weisse T, Colijn F, Veldhuis MJW, Davies A and Wassmann P (1987) *Phaeocystis* blooms and nutrient enrichment in the continental coastal zones of the North *Sea. Ambio* **16**, 38–46

Leese F, Kop A, Agrawal S and Held C (2008) Isolation and characterization of microsatellite markers from the marine isopods *Serolis paradoxa* and *Septemserolis septemcarinata* (Crustacea: Peracarida). *Molecular Ecology Resources*, 2008

Martins W, de Sousa D, Proite K, Guimaraes P, Moretzsohn M and Bertioli D (2006) New softwares for automated microsatellite marker development. *Nucleic Acids Res,* **34**, e31

Mayer C (2008) Phobos paper (in press).

Mullis K (1990) The unusual origin of the polymerase chain reaction. *Sci. Am.* **262**, 56–61

Mullis K, *et al*. System for automated performance of the polymerase chain reaction. U.S. patent no. 5656493.

Ousterhout JK (1990) Tcl: An Embeddable Command Language. Winter USENIX Proceedings, 1990.

Robinson AJ, Love CG, Batley J, Barker G and Edwards D (2004) Simple sequence repeat marker loci discovery using SSR primer. *Bioinformatics,* **20**, 1475

Rozen S and Skaletsky H (2000) Primer3 on the WWW for general users and for biologist programmers. *Methods Mol. Biol.*, **132**, 365-386

Schlötterer C (2004) The evolution of molecular markers - just a matter of fashion. *Nature Reviews Genetics,* **5**, 63-69

Shen P, van Rijssel M, Wang Y, Songhui L, Jufang C and Qi Y (2004) Toxic *Phaeocystis globosa* strain from China grows at remarkably high temperatures. In: **Steidinger KA, Landsberg JH, Tomas CR, Vargo GA** (eds) *Harmful Algae* 2002. Florida Fish and Wildlife Conservation Commission, Florida Institute of Oceanography and Intergovernmental Oceanographic Commission of UNESCO, St. Petersburg, 396–398

Smith TF and Waterman MS (1981) Identification of Common Molecular Subsequences. *Journal of molecular biology* London: Academic Press., **147**, 195-197

Staden R, Beal KF and Bonfield JK (1998). The Staden Package. In **Misener S & Krawetz SA (eds)**, Computer Methods in Molecular Biology, Bioinformatics Methods and Protocols. *The Humana Press Inc.*, Totowa, NJ, **132**, 115-130

Thurston MI and Field D (2005) Msatfinder: detection and characterisation of microsatellites. Distributed by the authors at [http://www.genomics.ceh.ac.uk/msatfinder/]

Turnpenny PD and Ellard S (2005) *Emery's Elements of Medical Genetics.* 12th edition. Edinburgh: Elsevier – Churchill Livingstone.

Veldhuis MJW and Wassmann P (2005) Bloom dynamics and biological control of a high biomass HAB species in European coastal waters: a *Phaeocystis* case study. *Harmful Algae* **4**, 805-809

Whipple SJ, Patten BC and Verity PG (2005) Life cycle of the marine alga *Phaeocystis*: a conceptual model to summarize literature and guide research*.* *J. Mar. Syst.* **57**, 83–110

Appendix

A Application of the project

A.1 Installation

To install the project you only have to execute the file „Multiplex_Project.exe“. You have to choose your „Staden Package“-folder for the installation, otherwise the installation won’t work.

CAUTION: In the process of installation the “gaprc”- and “GTAGDB”-files will be overwritten. If the installed version of Gap4 is already modified, it is possible that the modifications would not work after this installation. To avoid this problem the files should NOT be overwritten and the following line should be appended at the end of the “gaprc”-file manually:

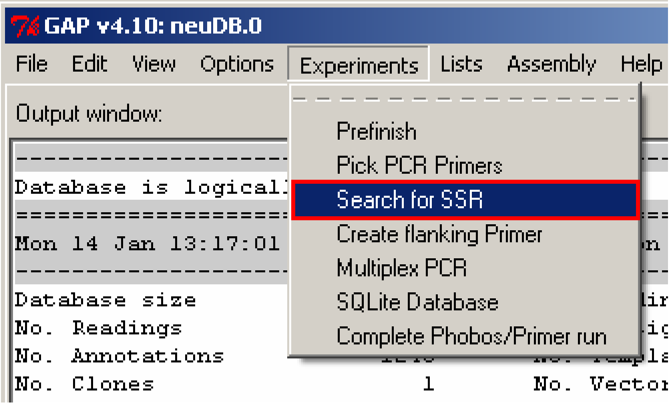
```
load_package %L/phobos_multiplex    {}     phobos_multiplex 0
```

The “GTAGDB”-file includes tag-definitions, the file will be overwritten to create parameters for the tags “REP1” to “REP9”, “PRI1” to “PRI9” and “MPL1” to “MPL9”. If the file already contains modifications, they will be removed after the file replacement.
“gaprc” and “GTAGDB” can be found in the “tables”-folder of the Staden Package.

A.2 Usage of the different modules

A.2.1 The Phobos Module

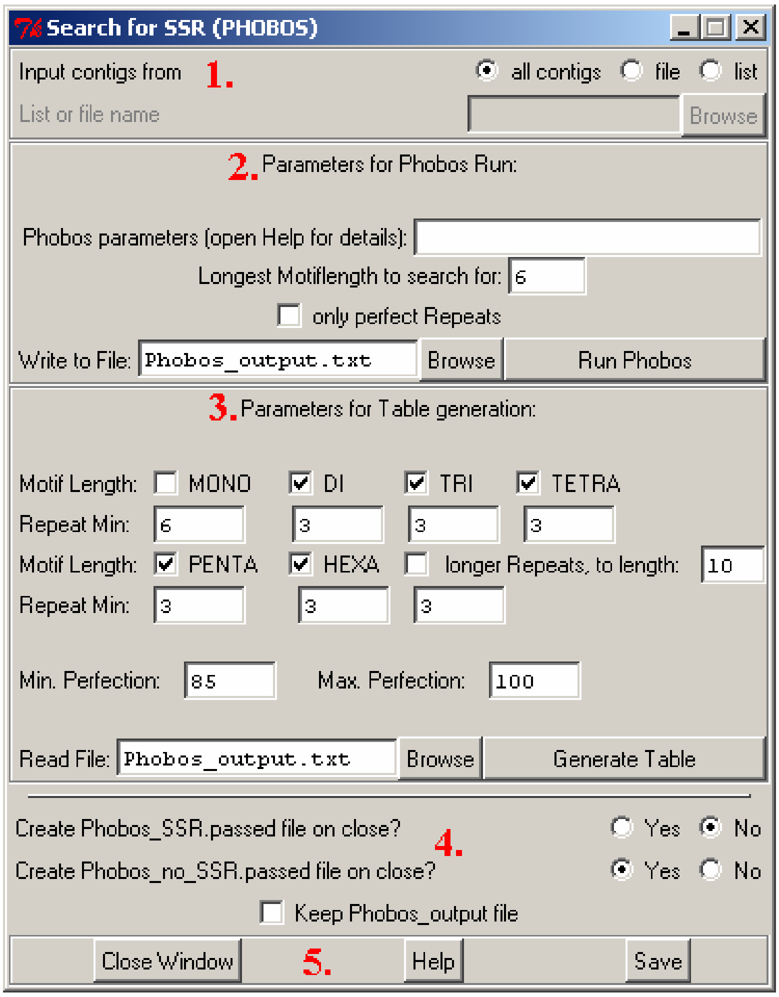
The Phobos Module searches for simple sequence repeats (SSR) in the sequences of the Gap4 database. These repeats could be annotated regions in the Gap4 database or stored in a table of a SQLite database (see section A.2.4)
The Phobos Module can be found under “Experiments” in the menulist. It is titled “Search for SSR”.



```
--------------------
Database is logical
====================
Mon 14 Jan 13:17:01
--------------------
Database size
No. Readings
No. Annotations
No. Clones                     1        No. Vector
```

A.2.1.1 The Interface

The image below shows the interface of the Phobos Module. The different components of the interface are marked with red numbers in this image. The function of these components will be explained in the following.



Search for SSR (PHOBOS)

Input contigs from 1. ● all contigs ○ file ○ list

List or file name Browse

2. Parameters for Phobos Run:

Phobos parameters (open Help for details):

Longest Motiflength to search for: 6

☐ only perfect Repeats

Write to File: Phobos_output.txt Browse Run Phobos

3. Parameters for Table generation:

Motif Length: ☐ MONO ☑ DI ☑ TRI ☑ TETRA

Repeat Min: 6 3 3 3

Motif Length: ☑ PENTA ☑ HEXA ☐ longer Repeats, to length: 10

Repeat Min: 3 3 3

Min. Perfection: 85 Max. Perfection: 100

Read File: Phobos_output.txt Browse Generate Table

Create Phobos_SSR.passed file on close? ○ Yes ● No

4.

Create Phobos_no_SSR.passed file on close? ● Yes ○ No

☐ Keep Phobos_output file

Close Window 5. Help Save

1. The contig input

The sequences from the Gap4 database for the repeat search can here be selected. The standard selection is “all contigs”, this means the whole database.

Under “file” a file could be loaded which has to contain a list of contigs from the database.

After clicking the “Browse”-button when the option “list” is selected a window with a list of different options will open. It can be chosen between “all contigs”, “all readings”, “contigs” and “readings”.

2. Parameters for Phobos Run

The text field “Phobos parameters” allows the setting of parameters for the repeat search, a list of the available parameters and their function can be found in “The Staden Multiplex Project.pdf” or the Phobos-Help page (button “Help”).

“Longest Motiflength to search for” refers to the maximal length of a repeat motif for which Phobos should search. With increasing motif length the time for the repeat search will also increase.

The text field “Write to File” sets the filename for the Phobos output. Press “Run Phobos” to start the microsatellite search.

3. Parameters for Table generation

The button “Generate Table” will create a list of all repeat which are in the Phobos output file and match the parameters set for the list content. With the checkboxes “MONO” to “HEXA” and “longer Repeats” the motif types of SSRs, which should be shown in the table, can be selected. The option “longer Repeats” only makes sense if the option “Longest Motiflength to search for” under *Point 2* was set higher then 6.

The text fields under the checkboxes allow a restriction of the minimal length of a SSR which are shown in the list. Also the minimal and maximal perfection for repeats in the list can be set.

Through the “Generate Table”-button any number of tables with different content can be created.

4. Phobos output and passed files

The „Phobos_SSR.passed“-file contains a list of contigs which include repeats. The „Phobos_no_SSR.passed“-file contains a list of contigs which does not include repeats. By loading the „Phobos_no_SSR.passed“-file (see section *Point 1*) a more specific SSR search could be run, which will be concentrated on contigs without obvious microsatellites.

5. The Buttons

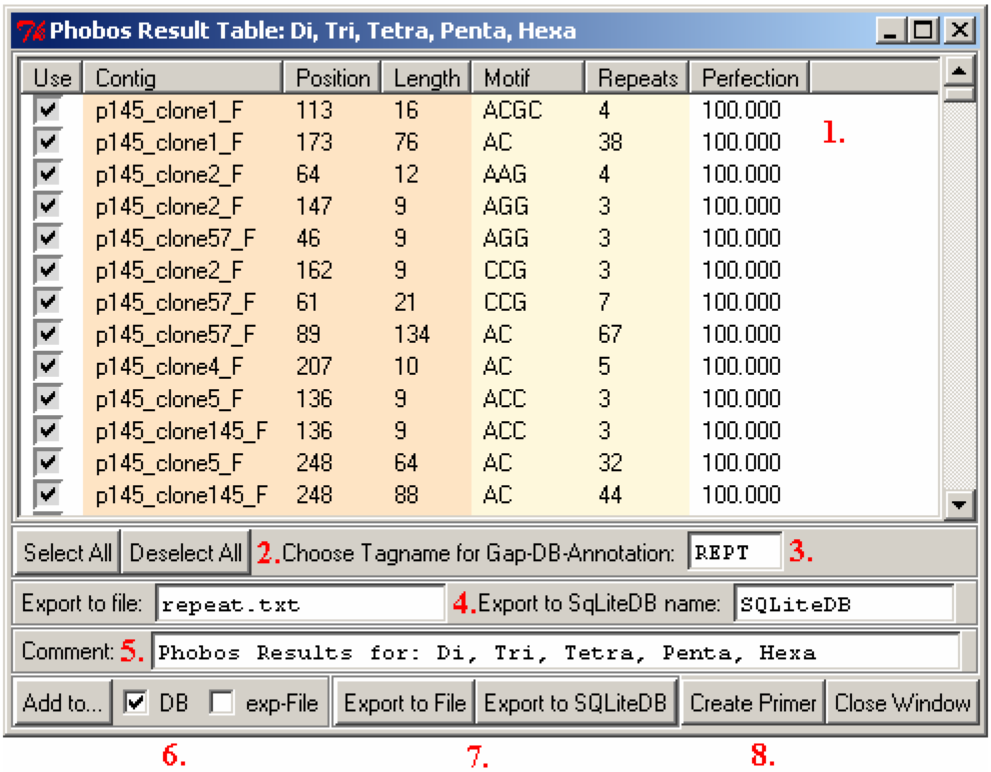
The “Close Window”-button will close the window and delete the Phobos_output file unless the “Keep Phobos_output file”-option under point 4 is selected.

The “Help”-button will open a HTML-page with information for handling the interface.

The “Safe”-button stores the interface parameters in the file “phobos_par.txt”, this file can be found in “{STADENROOT}\lib\phobos_multiplex\”.

A.2.1.2 The Repeat Table

The Repeat Table allows a manual selection of repeats for export to a tab-delimited text file or to the Gap4/SQLite database. The head of the table always shows which kinds of SSRs are contained in the list.



Phobos Result Table: Di, Tri, Tetra, Penta, Hexa

Use	Contig	Position	Length	Motif	Repeats	Perfection
☑	p145_clone1_F	113	16	ACGC	4	100.000
☑	p145_clone1_F	173	76	AC	38	100.000
☑	p145_clone2_F	64	12	AAG	4	100.000
☑	p145_clone2_F	147	9	AGG	3	100.000
☑	p145_clone57_F	46	9	AGG	3	100.000
☑	p145_clone2_F	162	9	CCG	3	100.000
☑	p145_clone57_F	61	21	CCG	7	100.000
☑	p145_clone57_F	89	134	AC	67	100.000
☑	p145_clone4_F	207	10	AC	5	100.000
☑	p145_clone5_F	136	9	ACC	3	100.000
☑	p145_clone145_F	136	9	ACC	3	100.000
☑	p145_clone5_F	248	64	AC	32	100.000
☑	p145_clone145_F	248	88	AC	44	100.000

Choose Tagname for Gap-DB-Annotation: REPT

Export to file: repeat.txt

Export to SqLiteDB name: SQLiteDB

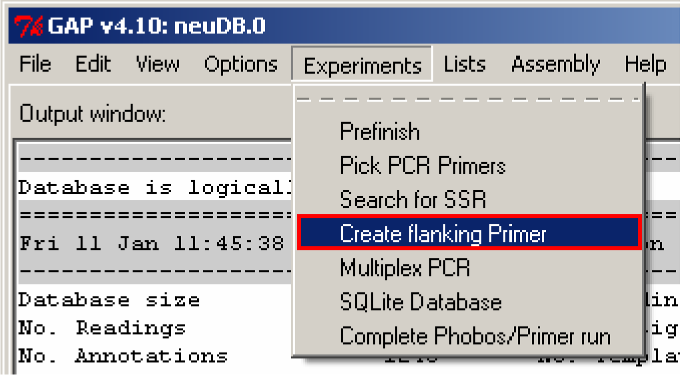
Comment: Phobos Results for: Di, Tri, Tetra, Penta, Hexa

☑ DB ☐ exp-File

1. A table of the repeats, extracted from the Phobos output file. A double click on a row opens a contig editor (see section A.3.1).
2. The buttons to select or deselect all repeats in the list.
3. The tag-name for the Gap4 database can be set here.
4. Here the names of the export file, SQLite database are set.
5. The comment set in this text field will be used for identification in the SQLite database.
6. The “Add to…”-button write all selected rows to either the Gap4 database or the experiment files, this depends on status of the checkboxes “DB” and “exp-File”.
7. Exports all selected rows to a text file (“Export to File”) or to a SQLite database (“Export to SQLite Database”).
8. The button “Create Primertable” opens the interface of the Primer3 Module and uses all selected rows in the repeat table as targets. After a run the Primer3 interface will not close. It always keeps the same target information. After changing the repeat selection it is necessary to open a new Primer3 interface by clicking the “Create Primertable”-button again.

A.2.2 The Primer3 Module

The Primer3 Module allows the selection of primer candidates for targets sequences, for example microsatellites. These targets could be annotated regions in the Gap4 database or stored in a table of a SQLite database (see section A.2.4)

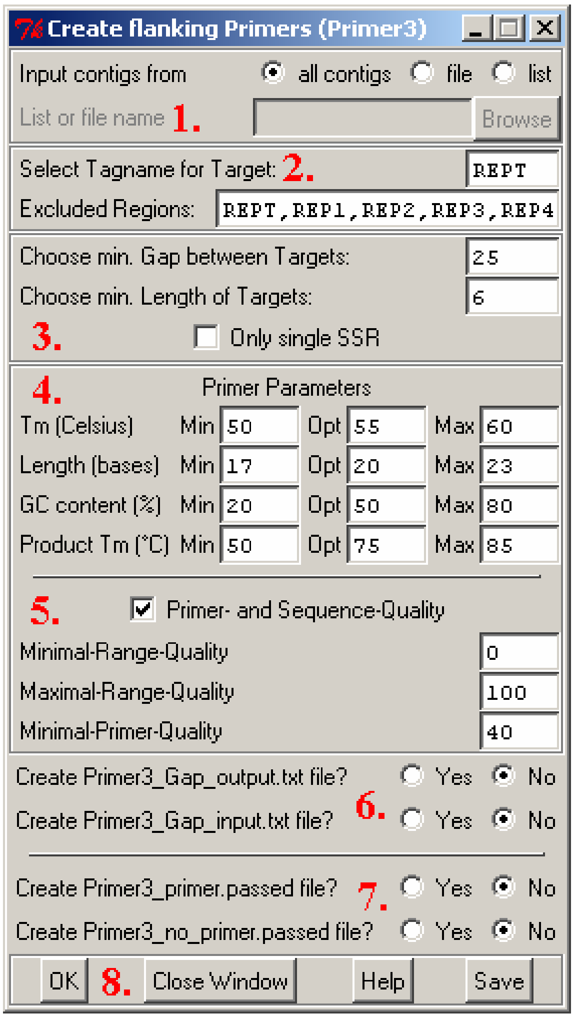


```
--------------------
Database is logical
====================
Fri 11 Jan 11:45:38
--------------------
Database size
No. Readings
No. Annotations
```

Like the other modules, the Primer3 Module can be found under "Experiments" in the menulist. It is titled "Create flanking Primer".

A.2.2.1 The Interface

The image below shows the interface of the Primer3 Module. The different components of the interface are marked with red numbers in this image. The function of these components will be explained in the following.



Input contigs from ● all contigs ○ file ○ list

List or file name

Select Tagname for Target: REPT

Excluded Regions: REPT,REP1,REP2,REP3,REP4

Choose min. Gap between Targets: 25

Choose min. Length of Targets: 6

□ Only single SSR

Primer Parameters

Tm (Celsius)	Min	50	Opt	55	Max	60
Length (bases)	Min	17	Opt	20	Max	23
GC content (%)	Min	20	Opt	50	Max	80
Product Tm (°C)	Min	50	Opt	75	Max	85

☑ Primer- and Sequence-Quality

Minimal-Range-Quality 0

Maximal-Range-Quality 100

Minimal-Primer-Quality 40

Create Primer3_Gap_output.txt file? ○ Yes ● No

Create Primer3_Gap_input.txt file? ○ Yes ● No

Create Primer3_primer.passed file? ○ Yes ● No

Create Primer3_no_primer.passed file? ○ Yes ● No

1. The contig input

The sequences from the Gap4 database for the primer search can here be selected. The standard selection is "all contigs", this means the whole database.

Under "file" a file could be loaded which has to contain a list of contigs from the database.

After clicking the "Browse"-button when the option "list" is selected a window with a list of different options will open. It can be chosen between "all contigs", "all readings", "contigs" and "readings".

2. Target- and excluded-regions

"Select Tagname for Target" sets a tag name which refers to the targets regions. The "Excluded Regions" text field should contain all tag names of marked regions were no primers should be placed, for example all repeat tag names (these are already set there as standard).

3. Setting of target parameters

The option “Choose min. Gap between Targets” has a default value of 25, this stands for the distance in bases which shall be between two targets. If two targets have less then this 25 bases between each other they will be counted as one. The default minimum length of a target is 6, only targets as long as the min. length or longer will be used for primer search.

The “Checkbox” with the option “Only single SSR” allow you only to search for targets of a single SSR-motif. (For example: if an AC-SSR lies next to AT-SSR it will be handled like one SSR and one target, if the “Only single SSR”-option is not selected. If the option is selected these SSRs won’t be counted as targets, because there is no room for a primer to be placed between them and the option will not allow two repeats in one primer product)

4. The primer parameters

Here some parameters can be set: the length of the primer in bases, the GC content, the melting temperature of the primer (“TM”) and the melting temperature of the product (“Product TM”).

5. Primer- and Sequence-Quality

If the “Primer- and Sequence-Quality”-checkbox is selected only sequences with quality values in between the “Minimal-Range-Quality” and the “Maximal-Range-Quality” are used for the primer search. Also every base of a primer location needs a minimal quality value like the one set at “Minimal-Primer-Quality”, otherwise the primer will not be created at this sequence position.

6. Primer3 input and output files

It can be chosen if the primer3 input and output text files should be kept or not, the default option is “No”.

7. Primer passed files

The „Primer3_primer.passed“-file contains a list of contigs which include primer for targets. The „Primer3_no_primer.passed“-file contains a list of contigs which does not include primers. By loading the „Primer3_no_primer.passed“-file (see *Point 1*) a more specific primer search could be run, which will be concentrated on contigs without useful primer locations.

8. The Buttons

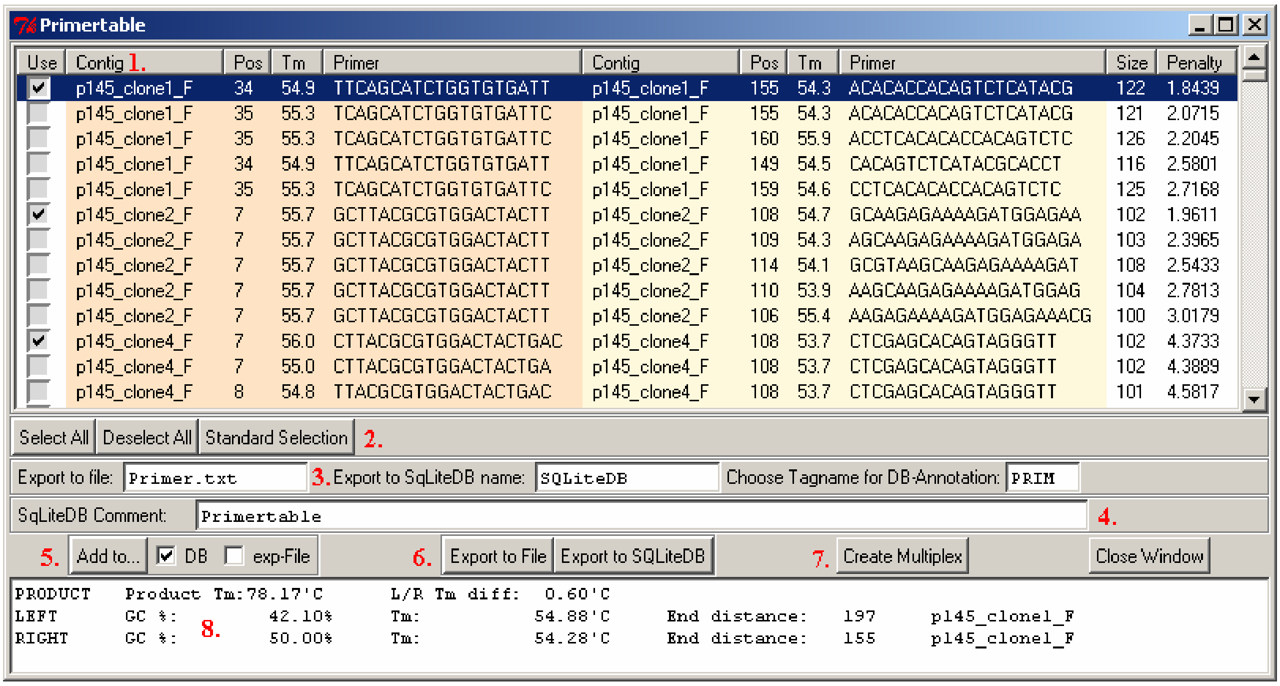
By clicking the “OK”-button Primer3 will be executed and a window opens which contains a table with possible primer candidates.

The “Help”-button opens a HTML-page with the information needed to handle the Primer3 interface.

The “Safe”-button stores the interface parameters in the file “primer3_par.txt”, it can be found in “{STADENROOT}\lib\phobos_multiplex\”.

A.2.2.2 The Primer table

The Primer Table allows a manual selection of primers for export to a tab-delimited text file or to the Gap4/SQLite database.



Use	Contig	Pos	Tm	Primer	Contig	Pos	Tm	Primer	Size	Penalty
☑	p145_clone1_F	34	54.9	TTCAGCATCTGGTGTGATT	p145_clone1_F	155	54.3	ACACACCACAGTCTCATACG	122	1.8439
□	p145_clone1_F	35	55.3	TCAGCATCTGGTGTGATTC	p145_clone1_F	155	54.3	ACACACCACAGTCTCATACG	121	2.0715
□	p145_clone1_F	35	55.3	TCAGCATCTGGTGTGATTC	p145_clone1_F	160	55.9	ACCTCACACACCACAGTCTC	126	2.2045
□	p145_clone1_F	34	54.9	TTCAGCATCTGGTGTGATT	p145_clone1_F	149	54.5	CACAGTCTCATACGCACCT	116	2.5801
□	p145_clone1_F	35	55.3	TCAGCATCTGGTGTGATTC	p145_clone1_F	159	54.6	CCTCACACACCACAGTCTC	125	2.7168
☑	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	108	54.7	GCAAGAGAAAAGATGGAGAA	102	1.9611
□	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	109	54.3	AGCAAGAGAAAAGATGGAGA	103	2.3965
□	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	114	54.1	GCGTAAGCAAGAGAAAAGAT	108	2.5433
□	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	110	53.9	AAGCAAGAGAAAAGATGGAG	104	2.7813
□	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	106	55.4	AAGAGAAAAGATGGAGAAACG	100	3.0179
☑	p145_clone4_F	7	56.0	CTTACGCGTGGACTACTGAC	p145_clone4_F	108	53.7	CTCGAGCACAGTAGGGTT	102	4.3733
□	p145_clone4_F	7	55.0	CTTACGCGTGGACTACTGA	p145_clone4_F	108	53.7	CTCGAGCACAGTAGGGTT	102	4.3889
□	p145_clone4_F	8	54.8	TTACGCGTGGACTACTGAC	p145_clone4_F	108	53.7	CTCGAGCACAGTAGGGTT	101	4.5817

Export to file: Primer.txt

Export to SqLiteDB name: SQLiteDB

Choose Tagname for DB-Annotation: PRIM

SqLiteDB Comment: Primertable

☑ DB □ exp-File

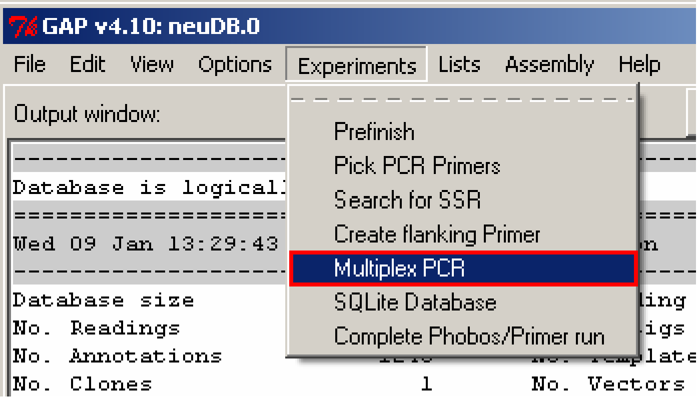
```
PRODUCT    Product Tm:78.17'C      L/R Tm diff:   0.60'C
LEFT       GC %:        42.10%     Tm:           54.88'C     End distance:    197     p145_clone1_F
RIGHT      GC %:        50.00%     Tm:           54.28'C     End distance:    155     p145_clone1_F
```

1. A table of the suggested primers, a double click on a row opens a contig editor (see section A.3.1).
2. The buttons to select or deselect all primers in the list, with the button “Standard Selection” the originally selection will be recreated.
3. Here the names of the export file, SQLite database and Gap4 database annotation are set.
4. The comment set in this text field will be used for identification in the SQLite database.
5. The “Add to…”-button write all selected rows to either the Gap4 database or the experiment files, this depends on status of the checkboxes “DB” and “exp-File”.
6. Exports all selected rows to a text file (“Export to File”) or to a SQLite database (“Export to SQLite Database”).

7. The button “Create Multiplex” opens the interface of the Multiplex module and uses all selected rows in the primer table for the creation of the multiplex. A second use of the “Create Multiplex”-option will destroy the multiplex table window and create a new one, so the multiplex information from the first run should be stored before starting a second run.
8. This text field shows additional primer information like the GC-content of the primer pair and the melting temperature of the primer product.

A.2.3 The Multiplex Module

The Multiplex module allows the selection of primer candidates, which will not interact with each other and so could be used in a multiplex-PCR.

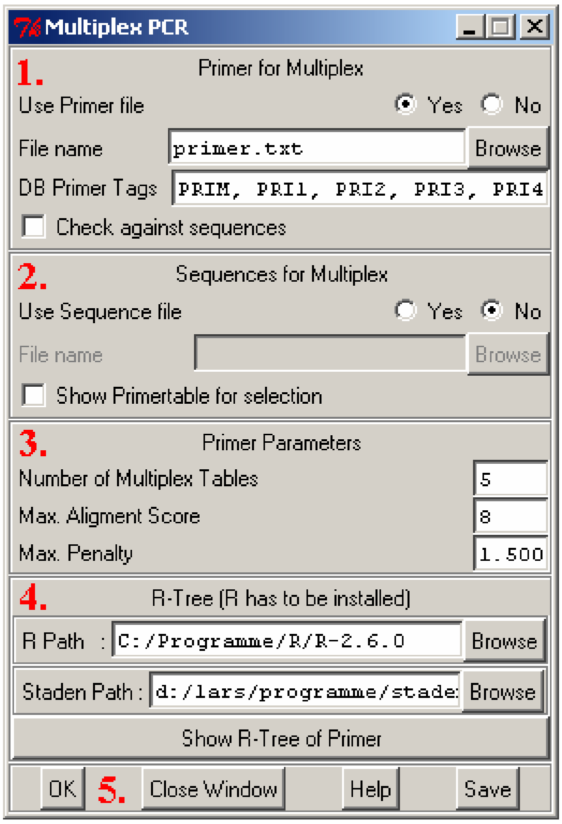


```
--------------------
Database is logical
====================
Wed 09 Jan 13:29:43                    n
--------------------
Database size                          ling
No. Readings                           igs
No. Annotations                        plate
No. Clones                  1         No. Vectors
```

Like the other modules, the Multiplex module can be found under “Experiments” in the menulist. It is titled “Multiplex PCR”.

A.2.3.1 The Interface

The image below shows the interface of the Multiplex module. The different components of the interface are marked with red numbers in this image. The function of these components will be explained in the following.



Use Primer file ● Yes ○ No
File name primer.txt
DB Primer Tags PRIM, PRI1, PRI2, PRI3, PRI4
□ Check against sequences
Use Sequence file ○ Yes ● No
File name
□ Show Primertable for selection
Number of Multiplex Tables 5
Max. Aligment Score 8
Max. Penalty 1.500
R Path : C:/Programme/R/R-2.6.0
Staden Path : d:/lars/programme/stade

1. Primer for Multiplex

In this component of the interface, the source of the primer information for the multiplex has to be set. If the option “Use Primer file” is set to “No” only the primers already written to the Gap4 database, with the tags set in “Primer Tags from DB”, will be used. Otherwise a file with primer information will be loaded. This should be a file created with the Primer3 Module (the “Experiments”-option: “Create flanking Primer”). Next there is a “Check against sequences”-checkbox. Only if this box is selected the interface part “Sequences for Multiplex” will be used.

2. Sequences for Multiplex

If the checkbox “Sequences for Multiplex” in Part 1 was selected the primers for the multiplex will be blasted against the sequences in the Gap4 database. As an alternative it is possible to load a FASTA file of sequences to blast against.

If the checkbox “Show Primertable for selection” is checked and the “OK”-Button in Part 5 is clicked a table will open, which allows a manual selection of the primer by means of the blast results for the creation of the multiplex (see section A.2.3.2).

3. Primer Parameters

Here will be set some parameters for the multiplex creation. The “Number of Multiplex Tables” sets the number of different multiplexes which will be created, with an increasing number the time for the calculation will also increase.

The “Max. Alignment Score” sets the maximal allowed number of matches between two primers to be included in one multiplex.

The “Max. Penalty” relates to the penalty value in the primer information, if a primer penalty higher than the set value the primer will not be used for the creation of a multiplex.

4. R-Tree

This component is only of interest, if the program “R” (http://www.r-project.org) is installed.

Under “R Path” the path to the version folder of the “R” program has to be set (see the interface image above for example). The “Staden Path” should already be set correctly.

After clicking the button “Show R-Tree of Primer” a PNG image of a tree of the primer pairs for the multiplex creation will be created an opened. The PNG will be created in the same folder as the currently opened Gap4 database. For more information about “R” and the primer pair tree see chapter 3.7 “The R-Project”.

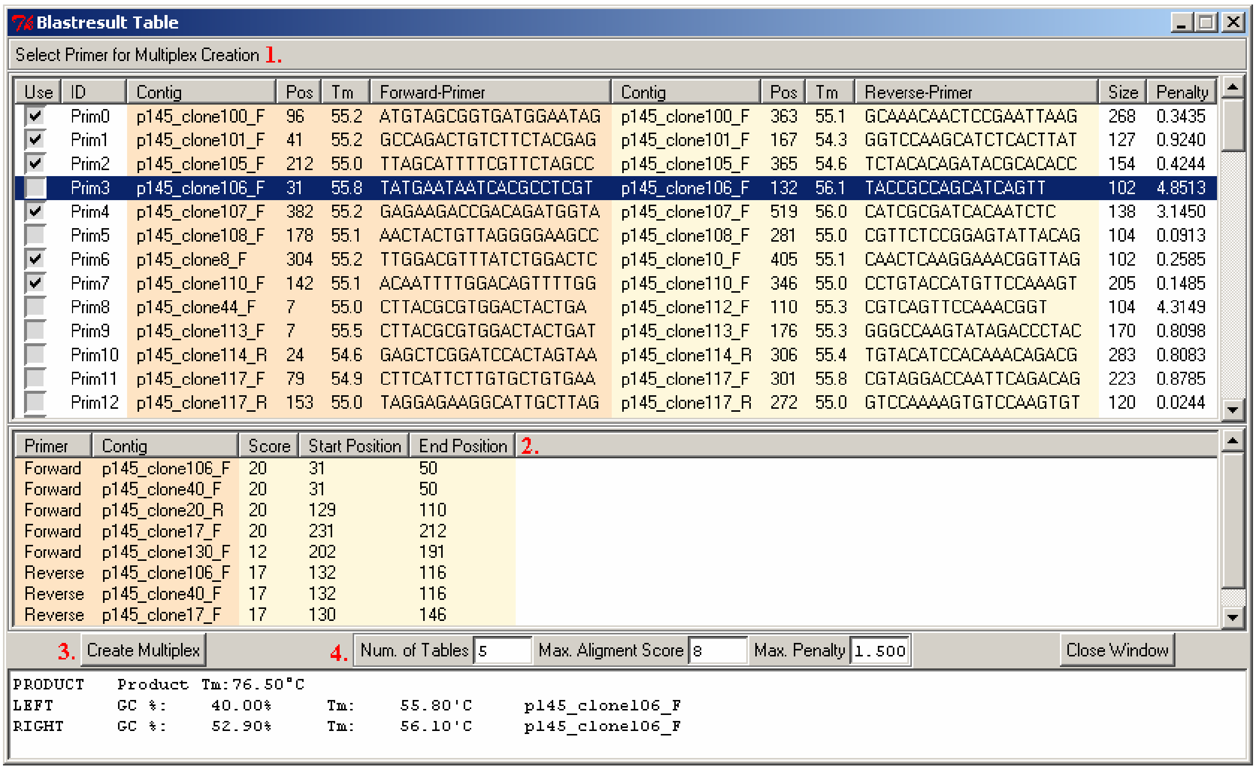
5. The Buttons

“OK” will open a blast-result table or the multiplex table, depending on the selection of the checkbox “Show Primertable for selection” in Part 2. “Help” opens a HTML-page with all information to the Multiplex module.

The “Safe”-button stores the interface parameters in the file “multiplex_par.txt”, it can be found in “{STADENROOT}\lib\phobos_multiplex\”.

A.2.3.2 The Blast-result table

The Blast-result Table allows a manual selection of primers as basis for the multiplex creation process.



Use	ID	Contig	Pos	Tm	Forward-Primer	Contig	Pos	Tm	Reverse-Primer	Size	Penalty
☑	Prim0	p145_clone100_F	96	55.2	ATGTAGCGGTGATGGAATAG	p145_clone100_F	363	55.1	GCAAACAACTCCGAATTAAG	268	0.3435
☑	Prim1	p145_clone101_F	41	55.2	GCCAGACTGTCTTCTACGAG	p145_clone101_F	167	54.3	GGTCCAAGCATCTCACTTAT	127	0.9240
☑	Prim2	p145_clone105_F	212	55.0	TTAGCATTTTCGTTCTAGCC	p145_clone105_F	365	54.6	TCTACACAGATACGCACACC	154	0.4244
☐	Prim3	p145_clone106_F	31	55.8	TATGAATAATCACGCCTCGT	p145_clone106_F	132	56.1	TACCGCCAGCATCAGTT	102	4.8513
☑	Prim4	p145_clone107_F	382	55.2	GAGAAGACCGACAGATGGTA	p145_clone107_F	519	56.0	CATCGCGATCACAATCTC	138	3.1450
☐	Prim5	p145_clone108_F	178	55.1	AACTACTGTTAGGGGAAGCC	p145_clone108_F	281	55.0	CGTTCTCCGGAGTATTACAG	104	0.0913
☑	Prim6	p145_clone8_F	304	55.2	TTGGACGTTTATCTGGACTC	p145_clone10_F	405	55.1	CAACTCAAGGAAACGGTTAG	102	0.2585
☑	Prim7	p145_clone110_F	142	55.1	ACAATTTTGGACAGTTTTGG	p145_clone110_F	346	55.0	CCTGTACCATGTTCCAAAGT	205	0.1485
☐	Prim8	p145_clone44_F	7	55.0	CTTACGCGTGGACTACTGA	p145_clone112_F	110	55.3	CGTCAGTTCCAAACGGT	104	4.3149
☐	Prim9	p145_clone113_F	7	55.5	CTTACGCGTGGACTACTGAT	p145_clone113_F	176	55.3	GGGCCAAGTATAGACCCTAC	170	0.8098
☐	Prim10	p145_clone114_R	24	54.6	GAGCTCGGATCCACTAGTAA	p145_clone114_R	306	55.4	TGTACATCCACAAACAGACG	283	0.8083
☐	Prim11	p145_clone117_F	79	54.9	CTTCATTCTTGTGCTGTGAA	p145_clone117_F	301	55.8	CGTAGGACCAATTCAGACAG	223	0.8785
☐	Prim12	p145_clone117_R	153	55.0	TAGGAGAAGGCATTGCTTAG	p145_clone117_R	272	55.0	GTCCAAAAGTGTCCAAGTGT	120	0.0244

Primer	Contig	Score	Start Position	End Position
Forward	p145_clone106_F	20	31	50
Forward	p145_clone40_F	20	31	50
Forward	p145_clone20_R	20	129	110
Forward	p145_clone17_F	20	231	212
Forward	p145_clone130_F	12	202	191
Reverse	p145_clone106_F	17	132	116
Reverse	p145_clone40_F	17	132	116
Reverse	p145_clone17_F	17	130	146

Num. of Tables 5 Max. Aligment Score 8 Max. Penalty 1.500

```
PRODUCT    Product  Tm:76.50°C
LEFT       GC %:    40.00%     Tm:    55.80'C     p145_clone106_F
RIGHT      GC %:    52.90%     Tm:    56.10'C     p145_clone106_F
```

1. A table with all primers from the primer file or the Gap4 database. Some primers are already selected on basis of the blast results. Whether a primer is selected or not depends only on the number of matches with the sequences of the Gap4 database or the loaded sequence file. Often the assembly of the Gap4 database is insufficient and the same sequence appears as a number of different contigs. In this case a primer could produce a number of hits, but actually has only a single binding place. Because of this, a manual selection or review is advisable.

   By a double click on a data row a contig editor will open, in which the primer product will be highlighted in the consensus.
2. Displays a table of all binding places of the selected primer pair. The first column of the table holds the information which of the two primers in a pair has a blast hit at the specific position; “Forward” refers to the forward primer of the selected pair and “Reverse” to the corresponding reverse primer. The column “Score” shows the number of

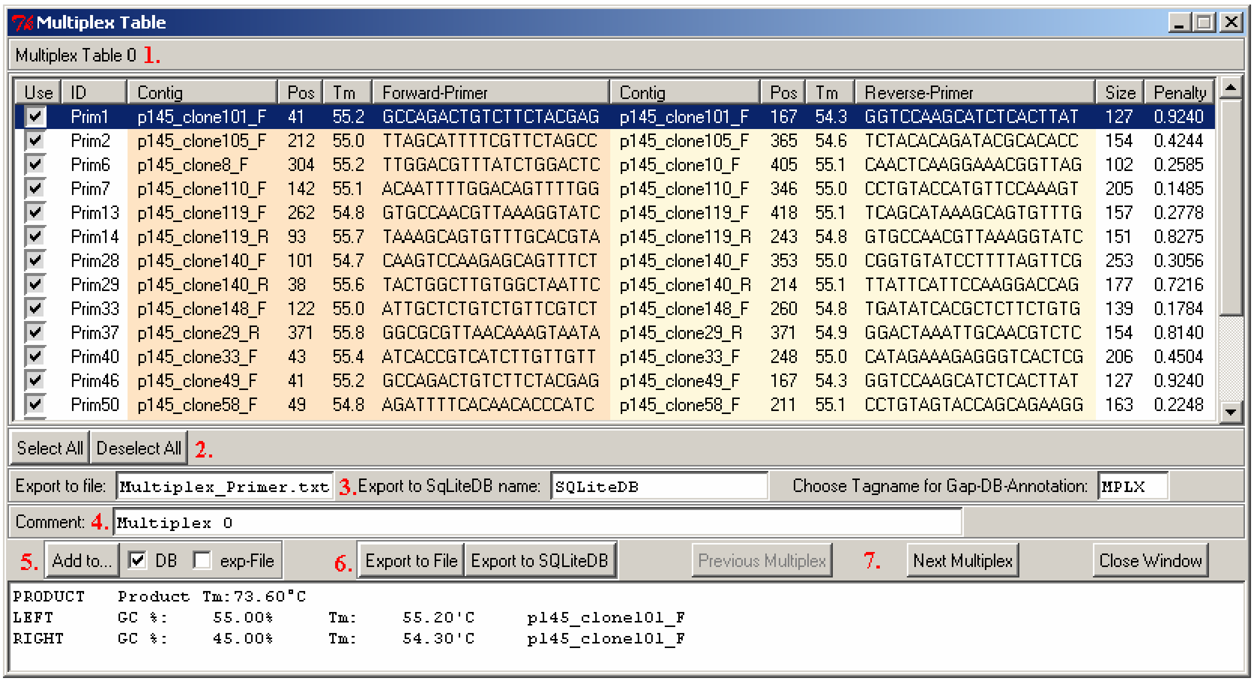
alignment matches for the blast hit. This means a score of 20 equals 20 bases of the primer which could anneal at this position of the sequence.

With a double click on a row in the table a contig editor will open, with a highlighted part in the consensus sequence.

3. The button “Create Multiplex”, which does exactly that, will open a new window with a table of proposed primer pairs to be used in a multiplex experimentally.
4. Shows the primer parameter earlier already set, here it is possible to change them again.

A.2.3.3 The Multiplex Table

The Multiplex Table displays all primer pairs useable in a single multiplex-PCR experiment.



Use	ID	Contig	Pos	Tm	Forward-Primer	Contig	Pos	Tm	Reverse-Primer	Size	Penalty
☑	Prim1	p145_clone101_F	41	55.2	GCCAGACTGTCTTCTACGAG	p145_clone101_F	167	54.3	GGTCCAAGCATCTCACTTAT	127	0.9240
☑	Prim2	p145_clone105_F	212	55.0	TTAGCATTTTCGTTCTAGCC	p145_clone105_F	365	54.6	TCTACACAGATACGCACACC	154	0.4244
☑	Prim6	p145_clone8_F	304	55.2	TTGGACGTTTATCTGGACTC	p145_clone10_F	405	55.1	CAACTCAAGGAAACGGTTAG	102	0.2585
☑	Prim7	p145_clone110_F	142	55.1	ACAATTTTGGACAGTTTTGG	p145_clone110_F	346	55.0	CCTGTACCATGTTCCAAAGT	205	0.1485
☑	Prim13	p145_clone119_F	262	54.8	GTGCCAACGTTAAAGGTATC	p145_clone119_F	418	55.1	TCAGCATAAAGCAGTGTTTG	157	0.2778
☑	Prim14	p145_clone119_R	93	55.7	TAAAGCAGTGTTTGCACGTA	p145_clone119_R	243	54.8	GTGCCAACGTTAAAGGTATC	151	0.8275
☑	Prim28	p145_clone140_F	101	54.7	CAAGTCCAAGAGCAGTTTCT	p145_clone140_F	353	55.0	CGGTGTATCCTTTTAGTTCG	253	0.3056
☑	Prim29	p145_clone140_R	38	55.6	TACTGGCTTGTGGCTAATTC	p145_clone140_R	214	55.1	TTATTCATTCCAAGGACCAG	177	0.7216
☑	Prim33	p145_clone148_F	122	55.0	ATTGCTCTGTCTGTTCGTCT	p145_clone148_F	260	54.8	TGATATCACGCTCTTCTGTG	139	0.1784
☑	Prim37	p145_clone29_R	371	55.8	GGCGCGTTAACAAAGTAATA	p145_clone29_R	371	54.9	GGACTAAATTGCAACGTCTC	154	0.8140
☑	Prim40	p145_clone33_F	43	55.4	ATCACCGTCATCTTGTTGTT	p145_clone33_F	248	55.0	CATAGAAAGAGGGTCACTCG	206	0.4504
☑	Prim46	p145_clone49_F	41	55.2	GCCAGACTGTCTTCTACGAG	p145_clone49_F	167	54.3	GGTCCAAGCATCTCACTTAT	127	0.9240
☑	Prim50	p145_clone58_F	49	54.8	AGATTTTCACAACACCCATC	p145_clone58_F	211	55.1	CCTGTAGTACCAGCAGAAGG	163	0.2248

Export to file: Multiplex_Primer.txt

Export to SqLiteDB name: SQLiteDB

Choose Tagname for Gap-DB-Annotation: MPLX

Comment: Multiplex 0

```
PRODUCT   Product  Tm:73.60'C
LEFT      GC %:     55.00%      Tm:     55.20'C     p145_clone101_F
RIGHT     GC %:     45.00%      Tm:     54.30'C     p145_clone101_F
```

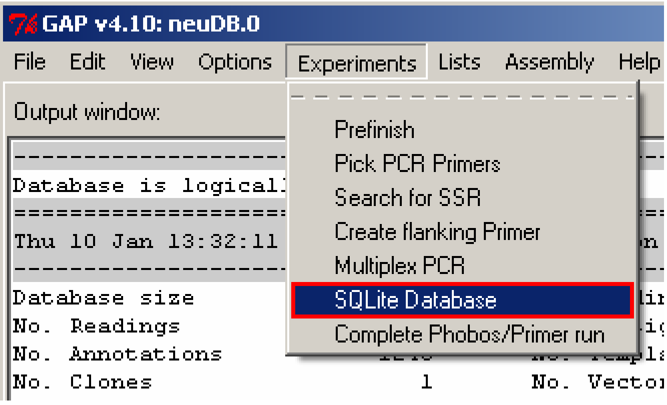
1. A primer table like the one in the blast-result table, but here only the compatible primer pairs are displayed.
2. The buttons under here enable to select/deselect all primers in the list.
3. Here the names of the export file, SQLite database and Gap4 database annotation are set.
4. The text field where a comment can be written, this comment allows for an easy identification of this table when written to the SQLite database and later reopened.
5. Here the names of the export file, SQLite database and Gap4 database annotation are set.
6. With the button “Safe Change” the comment for the table can be changed. The actual comment will be used if the table is again exported to a SQLite database.

7. These two buttons, “Previous Multiplex” and “Next Multiplex”, enable to go through the different multiplex tables which were created. The number of available tables depends of the number set at the parameter “Number of Tables”.

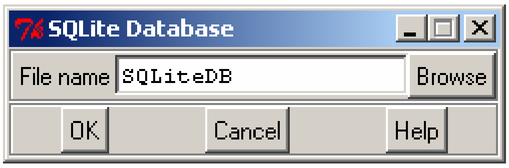
A.2.4 The SQLite Module

The SQLite Module allows the usage of a SQL database. With this database it is possible to safe all created repeat, primer and multiplex tables and use these tables again for later research.

Like the other modules, the SQLite Module can be found under “Experiments” in the menulist. It is titled “SQLite Database”.



```
--------------------
Database is logical
====================
Thu 10 Jan 13:32:11
--------------------
Database size
No. Readings
No. Annotations
No. Clones                    1      No. Vecto
```

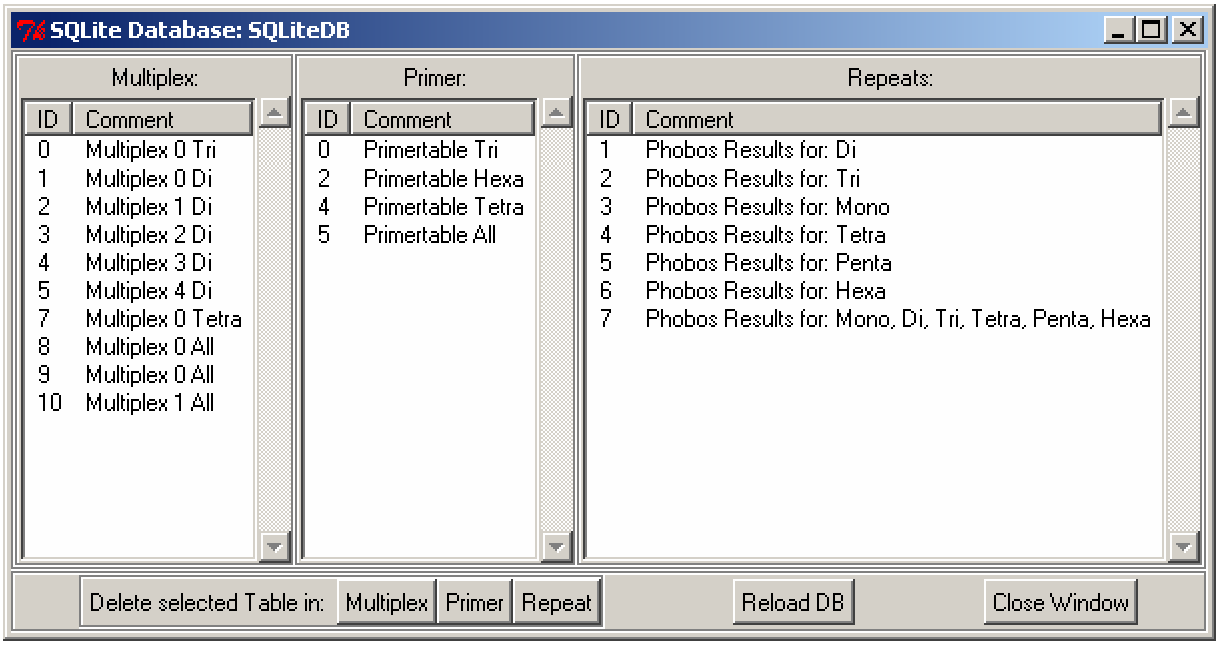


File name SQLiteDB

After clicking the “SQLite Database” menu option a little window will open for selection of the desired database.

A.2.4.1 The SQLite Database

When the database is opened a window like the one in the picture below will be created. This window consists of three lists, a multiplex-, a primer- and a repeat-list. Every list contains tables of according type. Every table in a list has a unique ID. If a primer table is written to the database it automatically will appear in the right list by the next opening of the database. With a double click on a row of a list the corresponding table will open.



Multiplex:

ID	Comment
0	Multiplex 0 Tri
1	Multiplex 0 Di
2	Multiplex 1 Di
3	Multiplex 2 Di
4	Multiplex 3 Di
5	Multiplex 4 Di
7	Multiplex 0 Tetra
8	Multiplex 0 All
9	Multiplex 0 All
10	Multiplex 1 All

Primer:

ID	Comment
0	Primertable Tri
2	Primertable Hexa
4	Primertable Tetra
5	Primertable All

Repeats:

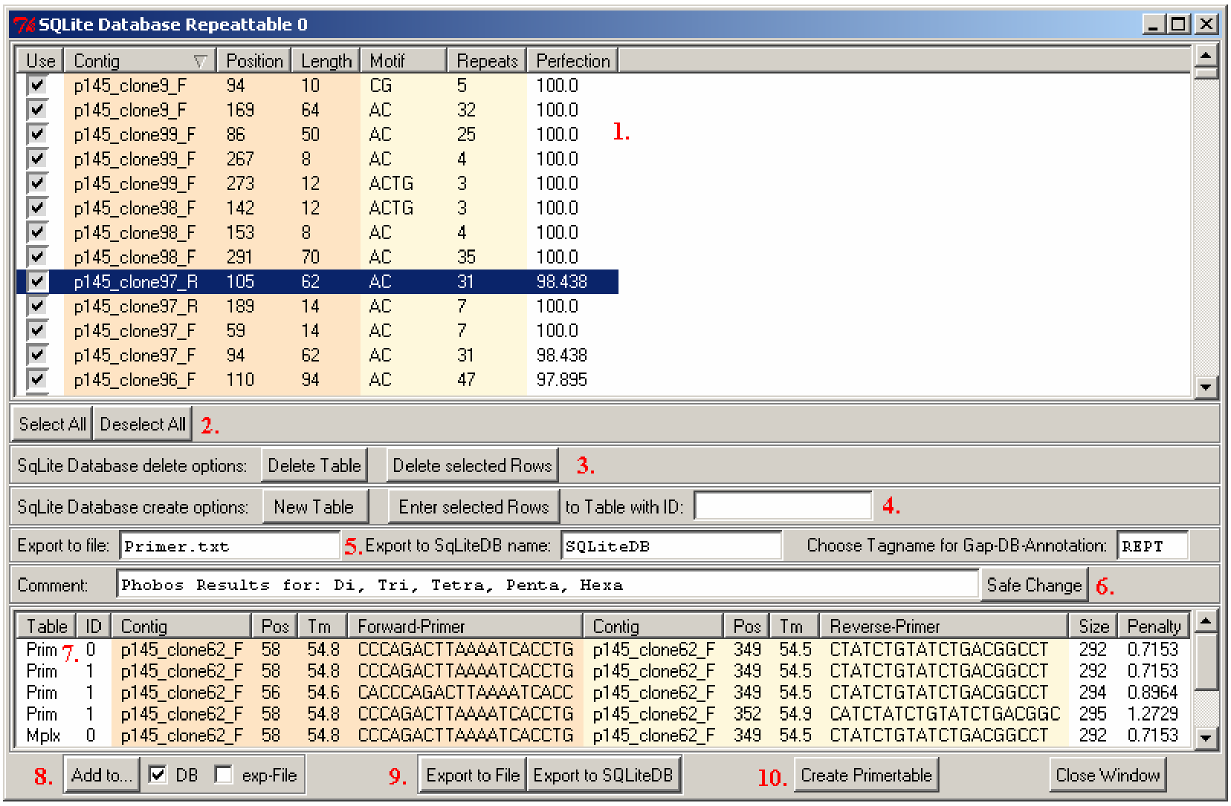
ID	Comment
1	Phobos Results for: Di
2	Phobos Results for: Tri
3	Phobos Results for: Mono
4	Phobos Results for: Tetra
5	Phobos Results for: Penta
6	Phobos Results for: Hexa
7	Phobos Results for: Mono, Di, Tri, Tetra, Penta, Hexa

Through the buttons “Multiplex”, “Primer” or “Repeat” it is possible to delete a selected table in the related list.

The button “Reload DB” allows to reload the SQLite database, this is necessary when a table is added through one of the other modules.

A.2.4.2 The SQLite Tables

There are three kinds of tables in the SQLite database; at first the repeat table will be explained:



Use	Contig	Position	Length	Motif	Repeats	Perfection
☑	p145_clone9_F	94	10	CG	5	100.0
☑	p145_clone9_F	169	64	AC	32	100.0
☑	p145_clone99_F	86	50	AC	25	100.0
☑	p145_clone99_F	267	8	AC	4	100.0
☑	p145_clone99_F	273	12	ACTG	3	100.0
☑	p145_clone98_F	142	12	ACTG	3	100.0
☑	p145_clone98_F	153	8	AC	4	100.0
☑	p145_clone98_F	291	70	AC	35	100.0
☑	p145_clone97_R	105	62	AC	31	98.438
☑	p145_clone97_R	189	14	AC	7	100.0
☑	p145_clone97_F	59	14	AC	7	100.0
☑	p145_clone97_F	94	62	AC	31	98.438
☑	p145_clone96_F	110	94	AC	47	97.895

SqLite Database delete options:

SqLite Database create options: to Table with ID:

Export to file: Primer.txt Export to SqLiteDB name: SQLiteDB Choose Tagname for Gap-DB-Annotation: REPT

Comment: Phobos Results for: Di, Tri, Tetra, Penta, Hexa

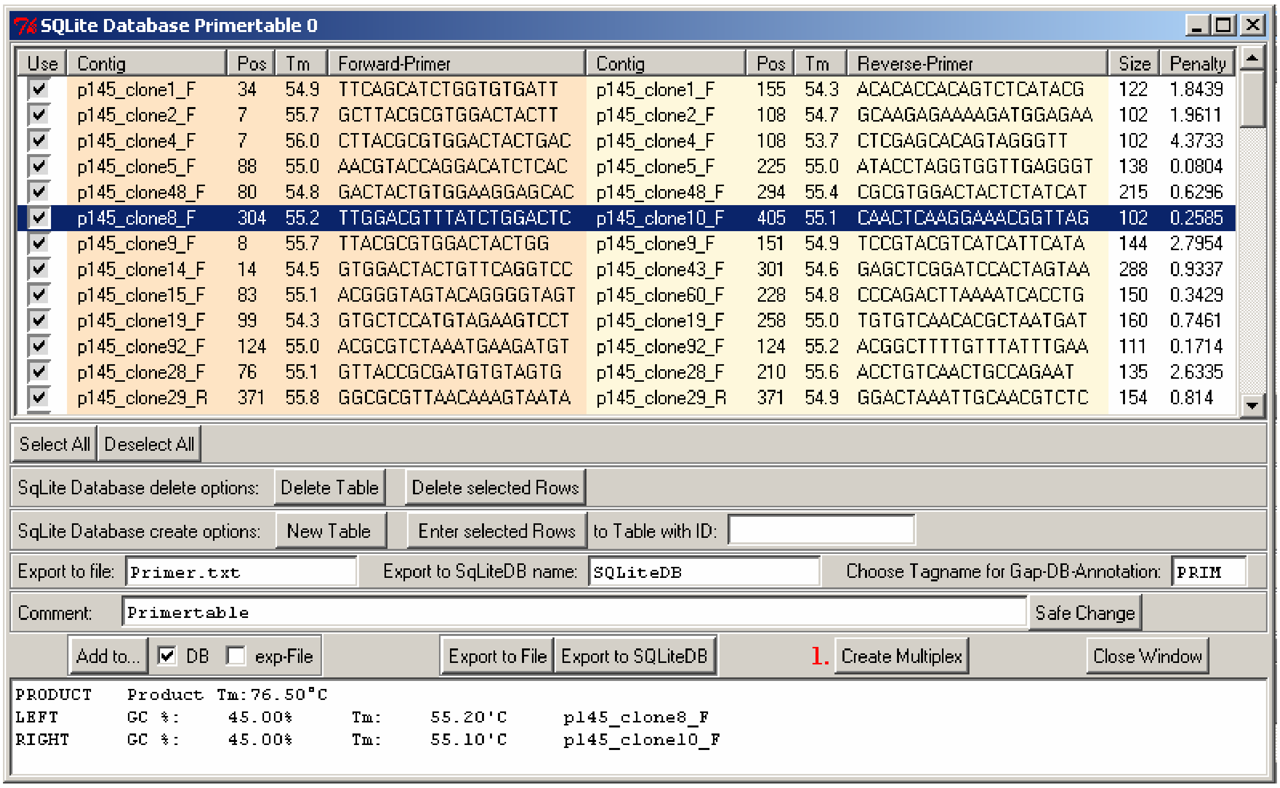
Table	ID	Contig	Pos	Tm	Forward-Primer	Contig	Pos	Tm	Reverse-Primer	Size	Penalty
Prim	0	p145_clone62_F	58	54.8	CCCAGACTTAAAATCACCTG	p145_clone62_F	349	54.5	CTATCTGTATCTGACGGCCT	292	0.7153
Prim	1	p145_clone62_F	58	54.8	CCCAGACTTAAAATCACCTG	p145_clone62_F	349	54.5	CTATCTGTATCTGACGGCCT	292	0.7153
Prim	1	p145_clone62_F	56	54.6	CACCCAGACTTAAAATCACC	p145_clone62_F	349	54.5	CTATCTGTATCTGACGGCCT	294	0.8964
Prim	1	p145_clone62_F	58	54.8	CCCAGACTTAAAATCACCTG	p145_clone62_F	352	54.9	CATCTATCTGTATCTGACGGC	295	1.2729
Mplx	0	p145_clone62_F	58	54.8	CCCAGACTTAAAATCACCTG	p145_clone62_F	349	54.5	CTATCTGTATCTGACGGCCT	292	0.7153

☑ DB ☐ exp-File

1. A table of the repeats, a double click on a row opens a contig editor.
2. The buttons to select or deselect all repeats in the list.
3. The button “Delete Table” will delete this table from the database when clicked. The button “Delete selected Rows” deletes all rows from this table which are checked in the “Use”-column.
4. The button “New Table” creates an empty table; it appears in the appropriate list of the SQLite database. With the button “Enter selected Rows” all rows which are checked in the “Use”-column can be written to the table with the ID set in the text field.
5. Here the names of the export file, SQLite database and Gap4 database annotation are set.
6. With the button “Safe Change” the comment for the table can be changed. The actual comment will be used if the table is again exported to a SQLite database.
7. Here a table shows all primers in the database which refer to the selected repeat. The columns “Table” and “ID” show the location of the specific primer in the database. Here again a double click will open a contig editor.
8. The “Add to…”-button write all selected rows to either the Gap4 database or the experiment files, this depends on status of the checkboxes “DB” and “exp-File”.
9. Exports all selected rows to a text file (“Export to File”) or to a SQLite database (“Export to SQLite Database”).
10. The button “Create Primertable” opens the interface of the Primer3 Module and uses all selected rows in the repeat table as targets. After a run the Primer3 interface will not close. It always keeps the same target information. After changing the repeat selection it is necessary to open a new Primer3 interface by clicking the “Create Primertable”-button again.

The primer table nearly equals the repeat table with two differences:

1. There is no “Create Primertable”-button but a “Create Multiplex”-button. This will open the Multiplex module interface; otherwise the handling of this function is the same.
2. There is no second table list included like for *Point 7* in the repeat table but a text field with some additional primer information.


SQLite Database Primertable 0

Use	Contig	Pos	Tm	Forward-Primer	Contig	Pos	Tm	Reverse-Primer	Size	Penalty
☑	p145_clone1_F	34	54.9	TTCAGCATCTGGTGTGATT	p145_clone1_F	155	54.3	ACACACCACAGTCTCATACG	122	1.8439
☑	p145_clone2_F	7	55.7	GCTTACGCGTGGACTACTT	p145_clone2_F	108	54.7	GCAAGAGAAAAGATGGAGAA	102	1.9611
☑	p145_clone4_F	7	56.0	CTTACGCGTGGACTACTGAC	p145_clone4_F	108	53.7	CTCGAGCACAGTAGGGTT	102	4.3733
☑	p145_clone5_F	88	55.0	AACGTACCAGGACATCTCAC	p145_clone5_F	225	55.0	ATACCTAGGTGGTTGAGGGT	138	0.0804
☑	p145_clone48_F	80	54.8	GACTACTGTGGAAGGAGCAC	p145_clone48_F	294	55.4	CGCGTGGACTACTCTATCAT	215	0.6296
☑	p145_clone8_F	304	55.2	TTGGACGTTTATCTGGACTC	p145_clone10_F	405	55.1	CAACTCAAGGAAACGGTTAG	102	0.2585
☑	p145_clone9_F	8	55.7	TTACGCGTGGACTACTGG	p145_clone9_F	151	54.9	TCCGTACGTCATCATTCATA	144	2.7954
☑	p145_clone14_F	14	54.5	GTGGACTACTGTTCAGGTCC	p145_clone43_F	301	54.6	GAGCTCGGATCCACTAGTAA	288	0.9337
☑	p145_clone15_F	83	55.1	ACGGGTAGTACAGGGGTAGT	p145_clone60_F	228	54.8	CCCAGACTTAAAATCACCTG	150	0.3429
☑	p145_clone19_F	99	54.3	GTGCTCCATGTAGAAGTCCT	p145_clone19_F	258	55.0	TGTGTCAACACGCTAATGAT	160	0.7461
☑	p145_clone92_F	124	55.0	ACGCGTCTAAATGAAGATGT	p145_clone92_F	124	55.2	ACGGCTTTTGTTTATTTGAA	111	0.1714
☑	p145_clone28_F	76	55.1	GTTACCGCGATGTGTAGTG	p145_clone28_F	210	55.6	ACCTGTCAACTGCCAGAAT	135	2.6335
☑	p145_clone29_R	371	55.8	GGCGCGTTAACAAAGTAATA	p145_clone29_R	371	54.9	GGACTAAATTGCAACGTCTC	154	0.814

Export to file: Primer.txt

Export to SqLiteDB name: SQLiteDB

Choose Tagname for Gap-DB-Annotation: PRIM

Comment: Primertable

```
PRODUCT    Product  Tm:76.50°C
LEFT       GC %:     45.00%     Tm:     55.20'C      p145_clone8_F
RIGHT      GC %:     45.00%     Tm:     55.10'C      p145_clone10_F
```

The multiplex table is of the same kind as the primer table, with one difference; there is no "Create Multiplex"-button.

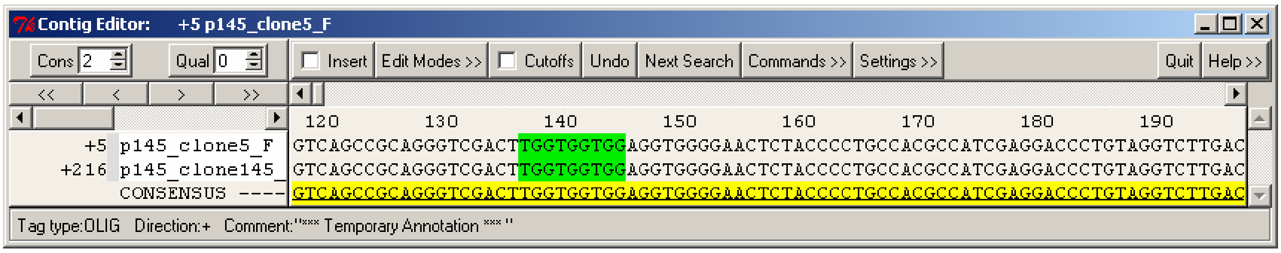
A.2.5 Export DB to FASTA

The option under "Phobos Multiplex Project" in the "Experiments"-menu, which is titled "Export to FASTA" creates a FASTA file of all contigs of the database. The file is created directly after clicking the option. The file is created in folder of the database and has the name "Gap4DB.fasta".

A.3 Additional Information

A.3.1 The Contig Editor

The contig editor shows the sequence of the selected contig. If it was opened from a table of one of the described modules it normally highlights in yellow a specific part of the sequence in the consensus line of the contig editor. In this example the product of a primer pair is highlighted, the microsatellite which is flanked by the primer pair can be seen highlighted in green in the contig sequences.

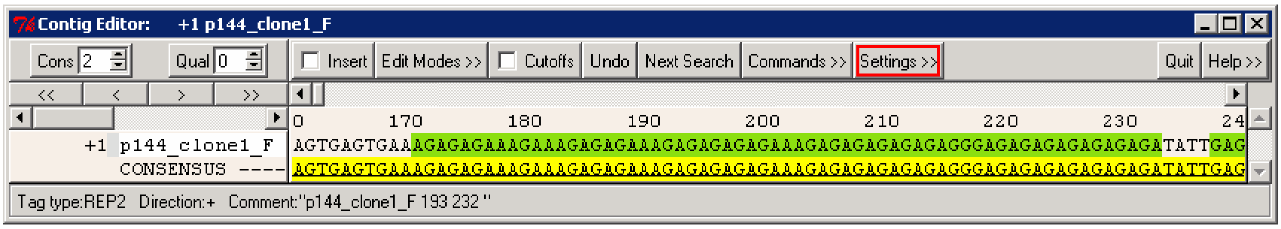


```
                      120       130       140       150       160       170       180       190
     +5 p145_clone5_F GTCAGCCGCAGGGTCGACTTGGTGGTGGAGGTGGGGAACTCTACCCCTGCCACGCCATCGAGGACCCTGTAGGTCTTGAC
   +216 p145_clone145_GTCAGCCGCAGGGTCGACTTGGTGGTGGAGGTGGGGAACTCTACCCCTGCCACGCCATCGAGGACCCTGTAGGTCTTGAC
        CONSENSUS ----GTCAGCCGCAGGGTCGACTTGGTGGTGGAGGTGGGGAACTCTACCCCTGCCACGCCATCGAGGACCCTGTAGGTCTTGAC
```

Tag type:OLIG Direction:+ Comment:"*** Temporary Annotation *** "

A.3.2 Useful information

Alternative to the standard tag-names “REPT” and “PRIM”, “REP1” to “REP9” and “PRI1” to “PRI9” can be used. Under the “Settings”-menu in the contig-editor the option “Set Active Tags” can be chosen. Here the tags which shall be marked in the contig can be selected; through this it is possible to get a clear view, if the contig contains many overlapping tags.

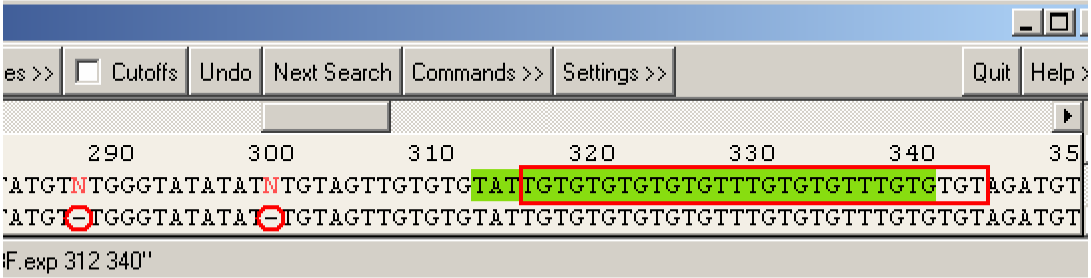


```
                    0     170       180       190       200       210       220       230       24
     +1 p144_clone1_F AGTGAGTGAAAGAGAGAAAGAAAGAGAGAAAGAGAGAGAGAAAGAGAGAGAGAGAGGGAGAGAGAGAGAGAGATATTGAG
        CONSENSUS ----AGTGAGTGAAAGAGAGAAAGAAAGAGAGAAAGAGAGAGAGAAAGAGAGAGAGAGAGGGAGAGAGAGAGAGAGATATTGAG
```

Tag type:REP2 Direction:+ Comment:"p144_clone1_F 193 232 "

Sometimes the repeat markings can be slightly displaced, like on the picture below. This offset is caused through gaps in the consensus sequence provoked by “N”s in the contig sequence.

The repeat position is generated on basis of the consensus sequence. In this sequence are no gaps. The gaps were created through the contig-editor by comparison with the original contig sequence. Therefore the consensus sequence seems longer in the contig-editor than it actually is, when used for repeat marking. Every gap in the sequence, which appears before a repeat tag, moves this tag one base to the left.



```
     290       300       310       320       330       340       35
ATGTNTGGGTATATATNTGTAGTTGTGTGTATTGTGTGTGTGTGTTTGTGTGTTTGTGTGTAGATGT
ATGT-TGGGTATATAT-TGTAGTTGTGTGTATTGTGTGTGTGTGTTTGTGTGTTTGTGTGTAGATGT
```

F.exp 312 340"

Acknowledgement

I thank Prof. Dr. Kauer for reviewing my diploma thesis and his openness for the topic of my project.

Further, I thank Prof. Dr. Stephan Frickenhaus, who made it possible for me to begin this project in the first place. He always supported me by the development of the project. Also I thank him for being the second examiner of this work.

I thank Dr. Bank Beszteri, who always had an answer for every theoretical or practical question that emerged. The project advanced decisively through his explanations.

Furthermore, I thank Dr. Christoph Mayer for allowing the usage of Phobos in this project and answering of any questions concerning it.

My sincere thanks to Steffi Gäbler-Schwarz, she made the laboratory part of this work possible. Her guidance in the different stages of the experiments was essential. I also want to thank Francisca Vermeulen and Helga Mehl for their help and patience in laboratory.

Florian Leese earns my thanks for testing the project and making his results available for me.

Thank for proof-reading of this work goes to Bank Beszteri, Stephan Frickenhaus and Rebekka Metzger. Also I thank Rebekka for her encouragement in all matters.

At last I want to thank my parents for standing behind me all of my life and my course of studies.