CLC **Genomics** Workbench

User manual

Manual for

CLC Genomics Workbench 4.5

Windows, Mac OS X and Linux

January 26, 2011

This software is for research purposes only.

CLC bio
Finlandsgade 10-12
DK-8200 Aarhus N
Denmark

Contents

Part I

Introduction

Chapter 1

Introduction to *CLC Genomics Workbench*

Contents

Welcome to *CLC Genomics Workbench* — a software package supporting your daily bioinformatics work.

We strongly encourage you to read this user manual in order to get the best possible basis for working with the software package.

This software is for research purposes only.

1.1 Contact information

The *CLC Genomics Workbench* is developed by:

CLC bio A/S
Science Park Aarhus
Finlandsgade 10-12
8200 Aarhus N
Denmark

http://www.clcbio.com

VAT no.: DK 28 30 50 87

Telephone: +45 70 22 55 09
Fax: +45 70 22 55 19

E-mail: info@clcbio.com

If you have questions or comments regarding the program, you are welcome to contact our support function:

E-mail: support@clcbio.com

1.2 Download and installation

The *CLC Genomics Workbench* is developed for Windows, Mac OS X and Linux. The software for either platform can be downloaded from http://www.clcbio.com/download.

Furthermore the program can be sent on a CD-Rom by regular mail. To receive the program by regular mail, please write an e-mail to support@clcbio.com, including your postal address.

1.2.1 Program download

The program is available for download on http://www.clcbio.com/download.

Before you download the program you are asked to fill in the **Download** dialog.

In the dialog you must choose:

- Which operating system you use
- Whether you would like to receive information about future releases

Depending on your operating system and your Internet browser, you are taken through some download options.

When the download of the installer (an application which facilitates the installation of the program) is complete, follow the platform specific instructions below to complete the installation procedure. [1]

[1] You must be connected to the Internet throughout the installation process.

1.2.2 Installation on Microsoft Windows

Starting the installation process is done in one of the following ways:

If you have downloaded an installer:
Locate the downloaded installer and double-click the icon.
The default location for downloaded files is your desktop.

If you are installing from a CD:
Insert the CD into your CD-ROM drive.

Choose the "Install *CLC Genomics Workbench*" from the menu displayed.

Installing the program is done in the following steps:

- On the welcome screen, click **Next**.
- Read and accept the License agreement and click **Next**.
- Choose where you would like to install the application and click **Next**.
- Choose a name for the Start Menu folder used to launch *CLC Genomics Workbench* and click **Next**.
- Choose if *CLC Genomics Workbench* should be used to open CLC files and click **Next**.
- Choose where you would like to create shortcuts for launching *CLC Genomics Workbench* and click **Next**.
- Choose if you would like to associate .clc files to *CLC Genomics Workbench*. If you check this option, double-clicking a file with a "clc" extension will open the *CLC Genomics Workbench*.
- Wait for the installation process to complete, choose whether you would like to launch *CLC Genomics Workbench* right away, and click **Finish**.

When the installation is complete the program can be launched from the Start Menu or from one of the shortcuts you chose to create.

1.2.3 Installation on Mac OS X

Starting the installation process is done in one of the following ways:

If you have downloaded an installer:
Locate the downloaded installer and double-click the icon.
The default location for downloaded files is your desktop.

If you are installing from a CD:
Insert the CD into your CD-ROM drive and open it by double-clicking on the CD icon on your desktop.

Launch the installer by double-clicking on the "*CLC Genomics Workbench*" icon.

Installing the program is done in the following steps:

- On the welcome screen, click **Next**.
- Read and accept the License agreement and click **Next**.
- Choose where you would like to install the application and click **Next**.
- Choose if *CLC Genomics Workbench* should be used to open CLC files and click **Next**.
- Choose whether you would like to create desktop icon for launching *CLC Genomics Workbench* and click **Next**.
- Choose if you would like to associate .clc files to *CLC Genomics Workbench*. If you check this option, double-clicking a file with a "clc" extension will open the *CLC Genomics Workbench*.
- Wait for the installation process to complete, choose whether you would like to launch *CLC Genomics Workbench* right away, and click **Finish**.

When the installation is complete the program can be launched from your Applications folder, or from the desktop shortcut you chose to create. If you like, you can drag the application icon to the dock for easy access.

1.2.4 Installation on Linux with an installer

Navigate to the directory containing the installer and execute it. This can be done by running a command similar to:

```
# sh CLCGenomicsWorkbench_4_JRE.sh
```

If you are installing from a CD the installers are located in the "linux" directory.

Installing the program is done in the following steps:

- On the welcome screen, click **Next**.
- Read and accept the License agreement and click **Next**.
- Choose where you would like to install the application and click **Next**.
  For a system-wide installation you can choose for example /opt or /usr/local. If you do not have root privileges you can choose to install in your home directory.
- Choose where you would like to create symbolic links to the program
  DO NOT create symbolic links in the same location as the application.
  Symbolic links should be installed in a location which is included in your environment PATH. For a system-wide installation you can choose for example /usr/local/bin. If you do not have root privileges you can create a 'bin' directory in your home directory and install symbolic links there. You can also choose not to create symbolic links.
- Wait for the installation process to complete and click **Finish**.

If you choose to create symbolic links in a location which is included in your PATH, the program can be executed by running the command:

```
# clcgenomicswb4
```

Otherwise you start the application by navigating to the location where you choose to install it and running the command:

```
# ./clcgenomicswb4
```

1.2.5 Installation on Linux with an RPM-package

Navigate to the directory containing the rpm-package and install it using the rpm-tool by running a command similar to:

```
# rpm -ivh CLCGenomicsWorkbench_4_JRE.rpm
```

If you are installing from a CD the rpm-packages are located in the "RPMS" directory.
Installation of RPM-packages usually requires root-privileges.

When the installation process is finished the program can be executed by running the command:

```
# clcgenomicswb4
```

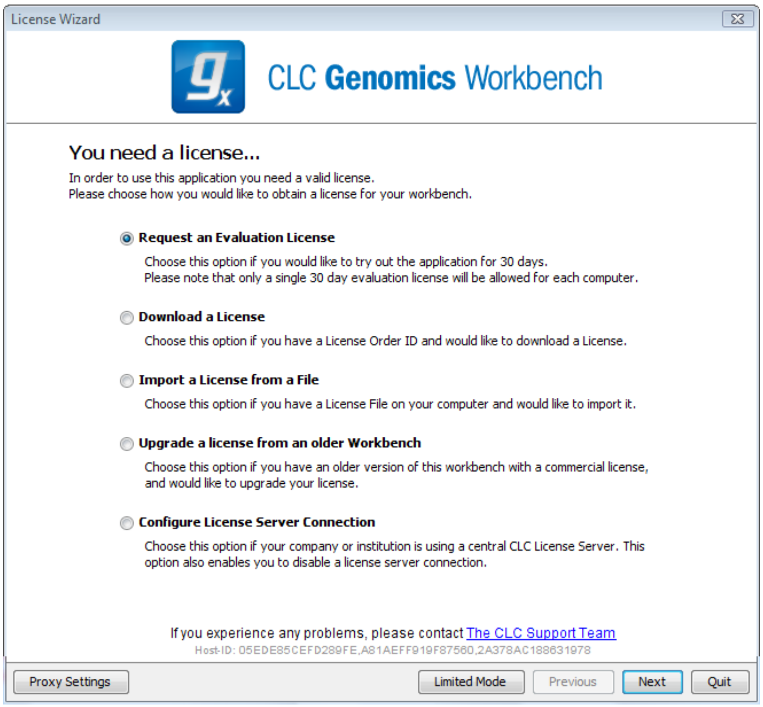
1.3 System requirements

- Windows 2000, Windows XP, Windows Vista or Windows 7
- Mac OS X 10.4 or newer
- Linux: Redhat or SuSE
- 256 MB RAM required
- 512 MB RAM recommended
- 1024 x 768 display recommended
- Intel or AMD CPU required
- **Small data sets**. Assembly and analysis of genomes up to 50 mega-bases and up to 10 mil. reads
  - 2 GB RAM required,
  - 4 GB RAM recommended
- **Medium data sets**. Assembly and analysis of larger genomes and up to 100 mil. reads
  - 8 GB RAM required,
  - 16 GB RAM recommended
- **Large data sets**. Assembly and analysis of larger genomes and more than 100 mil. reads
  - 16 GB RAM required,
  - 32 GB RAM recommended

- **Special requirements for de novo assembly**. De novo assembly may need more memory than stated above - this depends both on the number of reads and the complexity and size of the genome. See http://www.clcbio.com/white-paper for examples of the memory usage of various data sets.
- 64 bit computer and operating system required to use more than 2GB RAM

1.4 Licenses

When you have installed *CLC Genomics Workbench*, and start for the first time, you will meet the license assistant, shown in figure 1.1.


License Wizard

CLC Genomics Workbench

You need a license...

In order to use this application you need a valid license.
Please choose how you would like to obtain a license for your workbench.

● **Request an Evaluation License**

Choose this option if you would like to try out the application for 30 days.
Please note that only a single 30 day evaluation license will be allowed for each computer.

○ **Download a License**

Choose this option if you have a License Order ID and would like to download a License.

○ **Import a License from a File**

Choose this option if you have a License File on your computer and would like to import it.

○ **Upgrade a license from an older Workbench**

Choose this option if you have an older version of this workbench with a commercial license, and would like to upgrade your license.

○ **Configure License Server Connection**

Choose this option if your company or institution is using a central CLC License Server. This option also enables you to disable a license server connection.

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.1: *The license assistant showing you the options for getting started.*

The following options are available. They will be described in detail in the following sections.

- **Request an evaluation license**. The license is a fully functional, time-limited license (see below).
- **Download a license**. When you purchase a license, you will get a license ID from CLC bio. Using this option, you will get a license based on this ID.
- **Import a license from a file**. If CLC bio has provided a license file, or if you have downloaded a license from our web-based licensing system, you can import it using this option.
- **Upgrade license**. If you already have used a previous version of *CLC Genomics Workbench*, and you are entitled to upgrading to the new *CLC Genomics Workbench 4.5*, select this option to get a license upgrade.
- **Configure license server connection**. If your organization has a license server, select this option to connect to the server.

Select an appropriate option and click **Next**.

If for some reason you don't have access to getting a license, you can click the **Limited Mode** button (see section 1.4.6).

1.4.1 Request an evaluation license

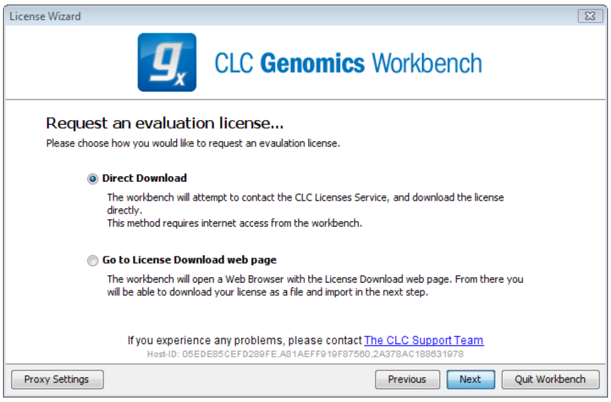
We offer a fully functional demo version of *CLC Genomics Workbench* to all users, free of charge.

Each user is entitled to 14 days demo of *CLC Genomics Workbench*. If you need more time for evaluating, another two weeks of demo can be requested.

We use the concept of *"quid quo pro"*. The last two weeks of free demo time given to you is therefore accompanied by a short-form questionnaire where you have the opportunity to give us feedback about the program.

The 30 days demo is offered for each major release of *CLC Genomics Workbench*. You will therefore have the opportunity to try the next major version when it is released. (If you purchase *CLC Genomics Workbench* the first year of updates is included.)

When you select to request an evaluation license, you will see the dialog shown in figure 1.2.



Request an evaluation license...

Please choose how you would like to request an evaulation license.

● Direct Download

The workbench will attempt to contact the CLC Licenses Service, and download the license directly.
This method requires internet access from the workbench.

○ Go to License Download web page

The workbench will open a Web Browser with the License Download web page. From there you will be able to download your license as a file and import in the next step.

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.2: *Choosing between direct download or download web page.*

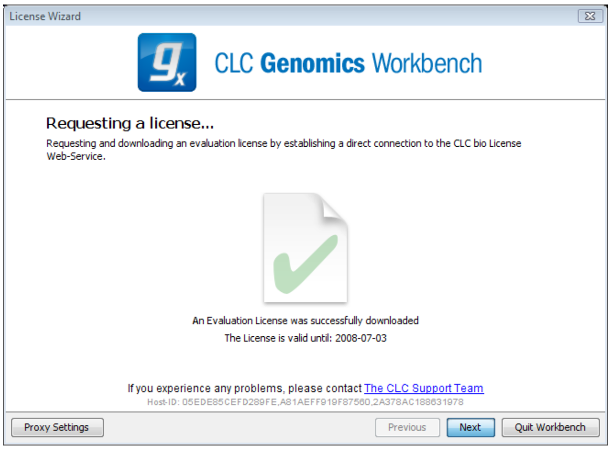
In this dialog, there are two options:

- **Direct download.** The workbench will attempt to contact the online CLC Licenses Service, and download the license directly. This method requires internet access from the workbench.
- **Go to license download web page.** The workbench will open a Web Browser with the License Download web page when you click **Next**. From there you will be able to download your license as a file and import it. This option allows you to get a license, even though the Workbench does not have direct access to the CLC Licenses Service.

If you select the first option, and it turns out that you do not have internet access from the Workbench (because of a firewall, proxy server etc.), you will be able to click **Previous** and use the other option instead.

Direct download

Selecting the first option takes you to the dialog shown in figure 1.3.



Requesting a license...

Requesting and downloading an evaluation license by establishing a direct connection to the CLC bio License Web-Service.

An Evaluation License was successfully downloaded

The License is valid until: 2008-07-03

If you experience any problems, please contact The CLC Support Team

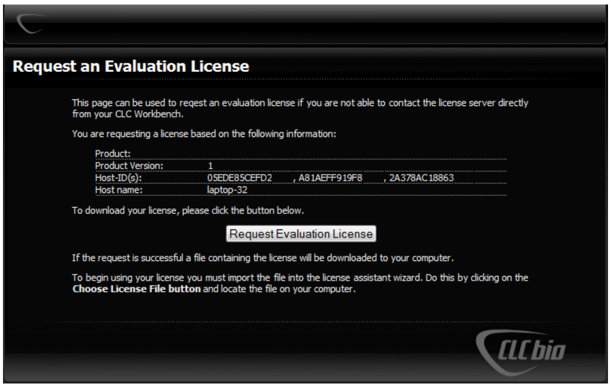
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.3: *A license has been downloaded.*

A progress for getting the license is shown, and when the license is downloaded, you will be able to click **Next**.

Go to license download web page

Selecting the second option, *Go to license download web page*, opens the license web page as shown in 1.4.



Request an Evaluation License

This page can be used to reqest an evaluation license if you are not able to contact the license server directly from your CLC Workbench.

You are requesting a license based on the following information:

Product:	
Product Version:	1
Host-ID(s):	05EDE85CEFD2 , A81AEFF919F8 , 2A378AC18863
Host name:	laptop-32

To download your license, please click the button below.

If the request is successful a file containing the license will be downloaded to your computer.

To begin using your license you must import the file into the license assistant wizard. Do this by clicking on the **Choose License File button** and locate the file on your computer.

Figure 1.4: *The license web page where you can download a license.*

Click the **Request Evaluation License** button, and you will be able to save the license on your computer, e.g. on the Desktop.

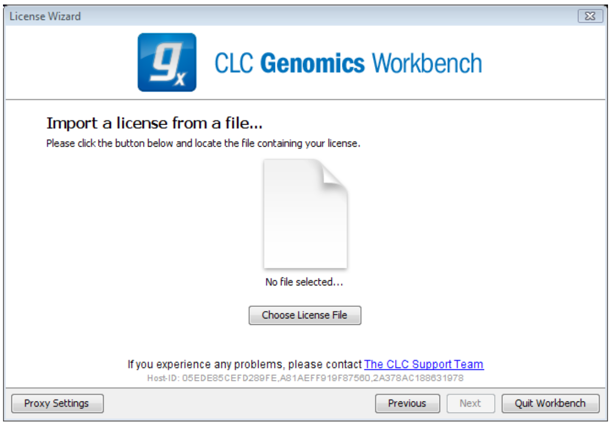
Back in the Workbench window, you will now see the dialog shown in 1.5.

Click the **Choose License File** button and browse to find the license file you saved before (e.g. on your Desktop). When you have selected the file, click **Next**.

Accepting the license agreement

Regardless of which option you chose above, you will now see the dialog shown in figure 1.6.

Please read the License agreement carefully before clicking **I accept these terms** and **Finish**.



Import a license from a file...

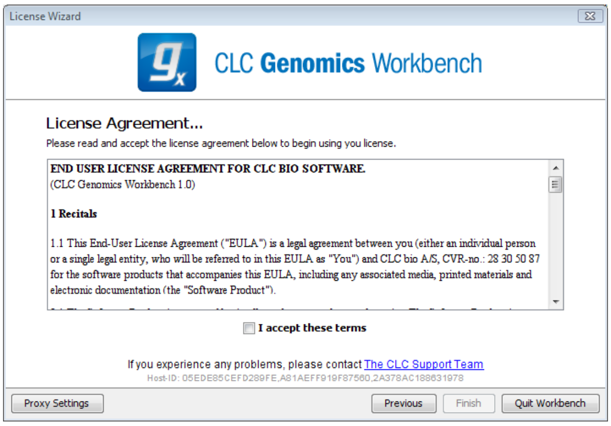
Please click the button below and locate the file containing your license.

No file selected...

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.5: *Importing the license downloaded from the web site.*



License Agreement...

Please read and accept the license agreement below to begin using you license.

END USER LICENSE AGREEMENT FOR CLC BIO SOFTWARE.
(CLC Genomics Workbench 1.0)

1 Recitals

1.1 This End-User License Agreement ("EULA") is a legal agreement between you (either an individual person or a single legal entity, who will be referred to in this EULA as "You") and CLC bio A/S, CVR-no.: 28 30 50 87 for the software products that accompanies this EULA, including any associated media, printed materials and electronic documentation (the "Software Product").

□ **I accept these terms**

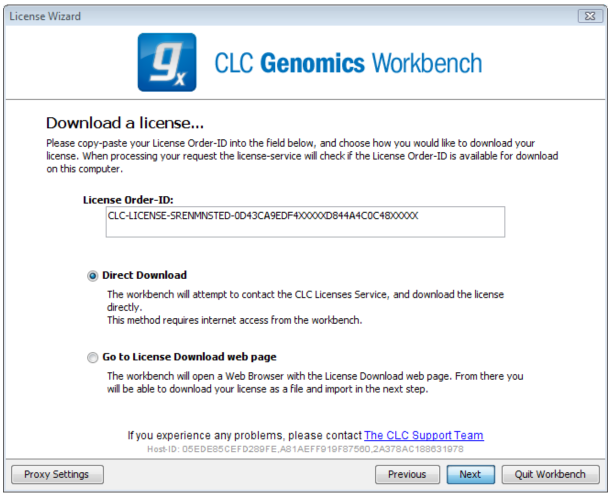
If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.6: *Read the license agreement carefully.*

1.4.2 Download a license

When you purchase a license, you will get a license ID from CLC bio. Using this option, you will get a license based on this ID. When you have clicked **Next**, you will see the dialog shown in 1.7. At the top, enter the ID (paste using Ctrl+V or ⌘ + V on Mac).



Download a license...

Please copy-paste your License Order-ID into the field below, and choose how you would like to download your license. When processing your request the license-service will check if the License Order-ID is available for download on this computer.

License Order-ID: CLC-LICENSE-SRENMNSTED-0D43CA9EDF4XXXXXD844A4C0C48XXXXX

● **Direct Download**

The workbench will attempt to contact the CLC Licenses Service, and download the license directly.
This method requires internet access from the workbench.

○ **Go to License Download web page**

The workbench will open a Web Browser with the License Download web page. From there you will be able to download your license as a file and import in the next step.

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.7: *Entering a license ID provided by CLC bio (the license ID in this example is artificial).*

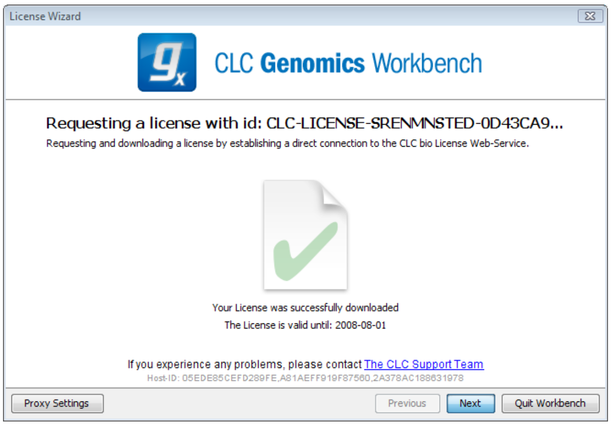
In this dialog, there are two options:

- **Direct download.** The workbench will attempt to contact the online CLC Licenses Service, and download the license directly. This method requires internet access from the workbench.
- **Go to license download web page.** The workbench will open a Web Browser with the License Download web page when you click **Next**. From there you will be able to download your license as a file and import it. This option allows you to get a license, even though the Workbench does not have direct access to the CLC Licenses Service.

If you select the first option, and it turns out that you do not have internet access from the Workbench (because of a firewall, proxy server etc.), you will be able to click **Previous** and use the other option instead.

Direct download

Selecting the first option takes you to the dialog shown in figure 1.8.



Requesting a license with id: CLC-LICENSE-SRENMNSTED-0D43CA9...

Requesting and downloading a license by establishing a direct connection to the CLC bio License Web-Service.

Your License was successfully downloaded

The License is valid until: 2008-08-01

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.8: *A license has been downloaded.*

A progress for getting the license is shown, and when the license is downloaded, you will be able to click **Next**.

Go to license download web page

Selecting the second option, *Go to license download web page*, opens the license web page as shown in 1.9.

Click the **Request Evaluation License** button, and you will be able to save the license on your computer, e.g. on the Desktop.

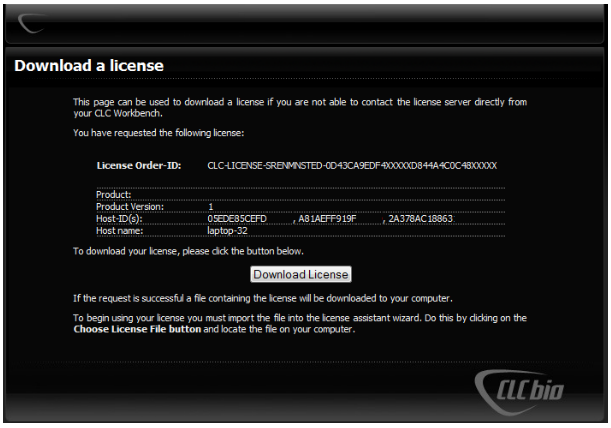
Back in the Workbench window, you will now see the dialog shown in 1.10.

Click the **Choose License File** button and browse to find the license file you saved before (e.g. on your Desktop). When you have selected the file, click **Next**.

Accepting the license agreement

Regardless of which option you chose above, you will now see the dialog shown in figure 1.11.

Please read the License agreement carefully before clicking **I accept these terms** and **Finish**.



Download a license

This page can be used to download a license if you are not able to contact the license server directly from your CLC Workbench.

You have requested the following license:

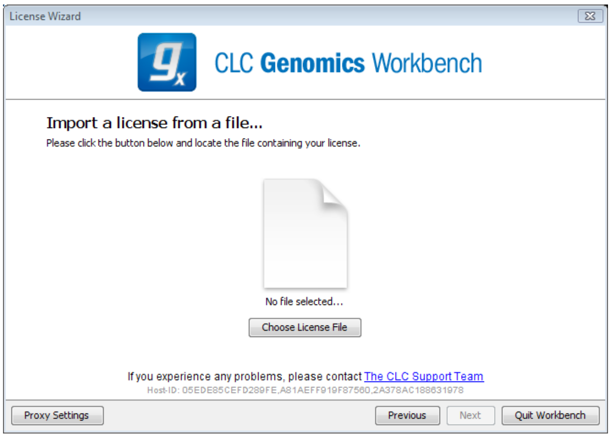
License Order-ID:	CLC-LICENSE-SRENMNSTED-0D43CA9EDF4XXXXXD844A4C0C48XXXXX
Product:	
Product Version:	1
Host-ID(s):	05EDE85CEFD , A81AEFF919F , 2A378AC18863
Host name:	laptop-32

To download your license, please click the button below.

If the request is successful a file containing the license will be downloaded to your computer.

To begin using your license you must import the file into the license assistant wizard. Do this by clicking on the **Choose License File button** and locate the file on your computer.

Figure 1.9: *The license web page where you can download a license.*



Import a license from a file...

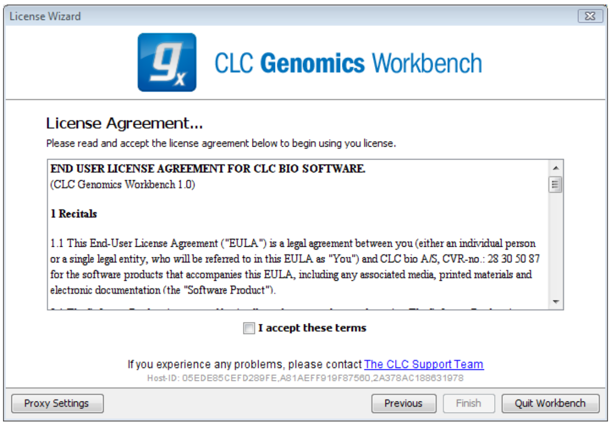
Please click the button below and locate the file containing your license.

No file selected...

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.10: *Importing the license downloaded from the web site.*



License Agreement...

Please read and accept the license agreement below to begin using you license.

END USER LICENSE AGREEMENT FOR CLC BIO SOFTWARE.
(CLC Genomics Workbench 1.0)

1 Recitals

1.1 This End-User License Agreement ("EULA") is a legal agreement between you (either an individual person or a single legal entity, who will be referred to in this EULA as "You") and CLC bio A/S, CVR-no.: 28 30 50 87 for the software products that accompanies this EULA, including any associated media, printed materials and electronic documentation (the "Software Product").

□ **I accept these terms**

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

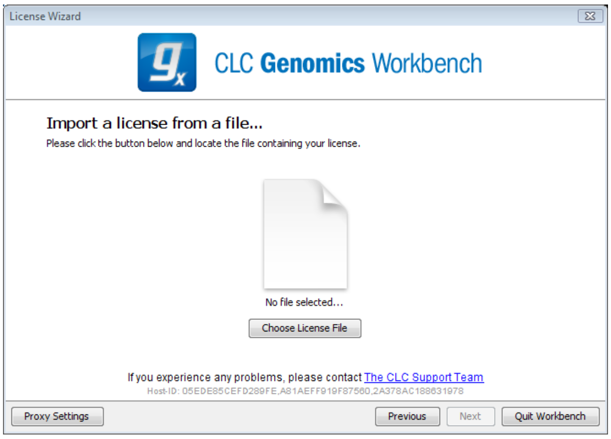
Figure 1.11: *Read the license agreement carefully.*

1.4.3 Import a license from a file

If you are provided a license file instead of a license ID, you will be able to import the file using this option.

When you have clicked **Next**, you will see the dialog shown in 1.12.

Click the **Choose License File** button and browse to find the license file provided by CLC bio. When you have selected the file, click **Next**.



License Wizard

CLC Genomics Workbench

Import a license from a file...

Please click the button below and locate the file containing your license.

No file selected...

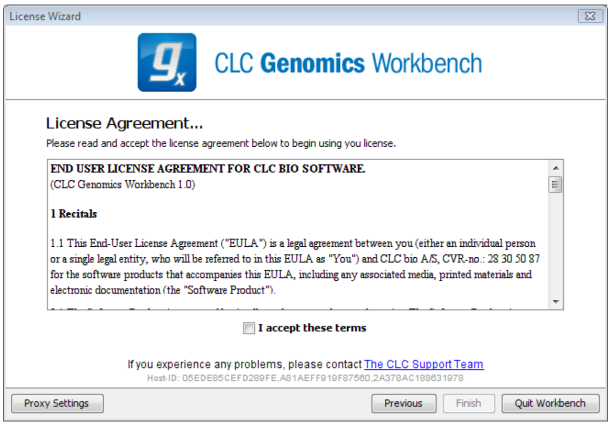
If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.12: *Selecting a license file .*

Accepting the license agreement

Regardless of which option you chose above, you will now see the dialog shown in figure 1.13.



License Wizard

CLC Genomics Workbench

License Agreement...

Please read and accept the license agreement below to begin using you license.

END USER LICENSE AGREEMENT FOR CLC BIO SOFTWARE.
(CLC Genomics Workbench 1.0)

1 Recitals

1.1 This End-User License Agreement ("EULA") is a legal agreement between you (either an individual person or a single legal entity, who will be referred to in this EULA as "You") and CLC bio A/S, CVR-no.: 28 30 50 87 for the software products that accompanies this EULA, including any associated media, printed materials and electronic documentation (the "Software Product").

□ **I accept these terms**

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.13: *Read the license agreement carefully.*

Please read the License agreement carefully before clicking **I accept these terms** and **Finish**.

1.4.4 Upgrade license

If you already have used a previous version of *CLC Genomics Workbench*, and you are entitled to upgrading to the new *CLC Genomics Workbench 4.5*, select this option to get a license upgrade.

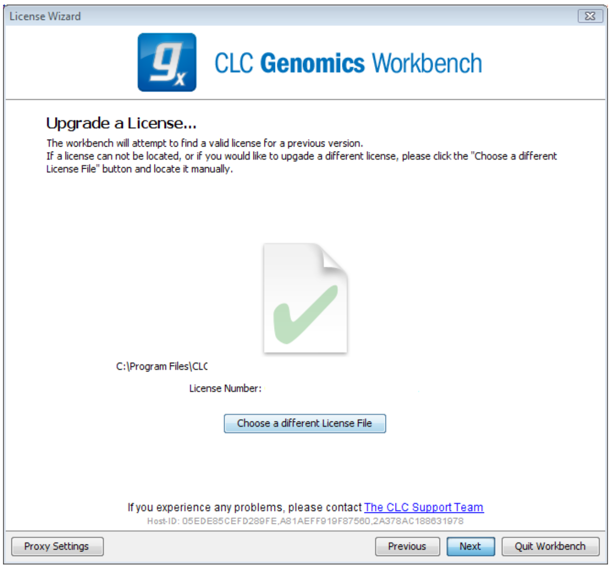
When you click **Next**, the workbench will search for a previous installation of *CLC Genomics Workbench*. It will then locate the old license.

If the Workbench succeeds to find an existing license, the next dialog will look as shown in figure 1.14.

When you click **Next**, the Workbench checks on CLC bio's web server to see if you are entitled to upgrade your license.

Note! If you should be entitled to get an upgrade, and you do not get one automatically in this process, please contact support@clcbio.com.

In this dialog, there are two options:



License Wizard

CLC Genomics Workbench

Upgrade a License...

The workbench will attempt to find a valid license for a previous version.
If a license can not be located, or if you would like to upgade a different license, please click the "Choose a different License File" button and locate it manually.

C:\Program Files\CLC

License Number:

If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

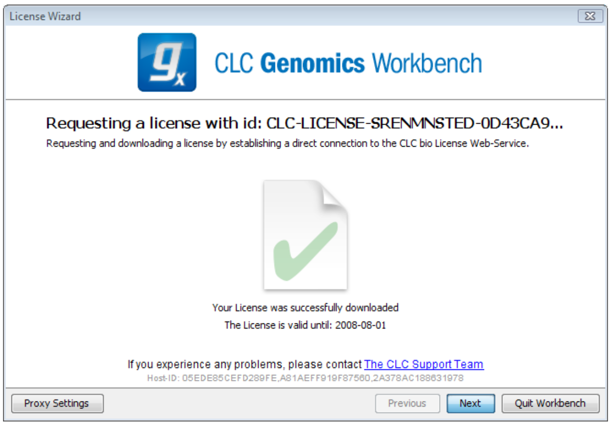
Figure 1.14: *An old license is detected.*

- **Direct download.** The workbench will attempt to contact the online CLC Licenses Service, and download the license directly. This method requires internet access from the workbench.
- **Go to license download web page.** The workbench will open a Web Browser with the License Download web page when you click **Next**. From there you will be able to download your license as a file and import it. This option allows you to get a license, even though the Workbench does not have direct access to the CLC Licenses Service.

If you select the first option, and it turns out that you do not have internet access from the Workbench (because of a firewall, proxy server etc.), you will be able to click **Previous** and use the other option instead.

Direct download

Selecting the first option takes you to the dialog shown in figure 1.15.



License Wizard

CLC Genomics Workbench

Requesting a license with id: CLC-LICENSE-SRENMNSTED-0D43CA9...

Requesting and downloading a license by establishing a direct connection to the CLC bio License Web-Service.

Your License was successfully downloaded
The License is valid until: 2008-08-01

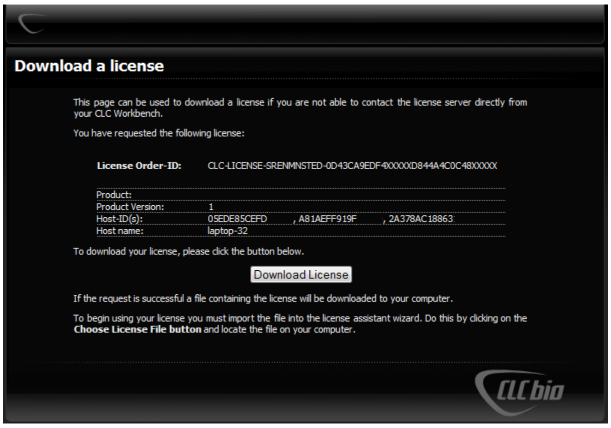
If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.15: *A license has been downloaded.*

A progress for getting the license is shown, and when the license is downloaded, you will be able to click **Next**.

Go to license download web page

Selecting the second option, *Go to license download web page*, opens the license web page as shown in 1.16.



Download a license

This page can be used to download a license if you are not able to contact the license server directly from your CLC Workbench.

You have requested the following license:

License Order-ID:	CLC-LICENSE-SRENMNSTED-0D43CA9EDF4XXXXXD844A4C0C48XXXXX
Product:	
Product Version:	1
Host-ID(s):	05EDE85CEFD , A81AEFF919F , 2A378AC18863
Host name:	laptop-32

To download your license, please click the button below.

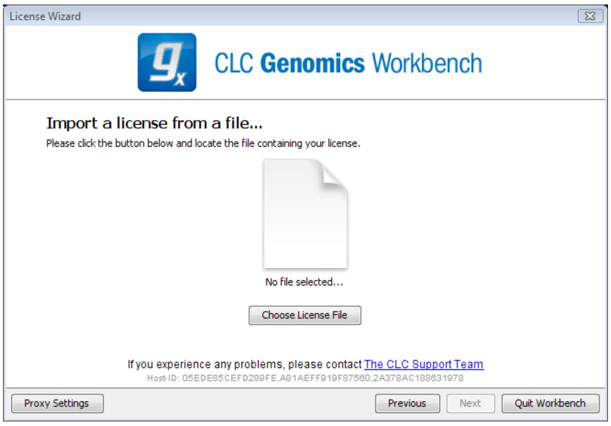
If the request is successful a file containing the license will be downloaded to your computer.

To begin using your license you must import the file into the license assistant wizard. Do this by clicking on the **Choose License File button** and locate the file on your computer.

Figure 1.16: *The license web page where you can download a license.*

Click the **Request Evaluation License** button, and you will be able to save the license on your computer, e.g. on the Desktop.

Back in the Workbench window, you will now see the dialog shown in 1.17.



License Wizard

CLC Genomics Workbench

Import a license from a file...

Please click the button below and locate the file containing your license.

No file selected...

If you experience any problems, please contact The CLC Support Team

Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.17: *Importing the license downloaded from the web site.*

Click the **Choose License File** button and browse to find the license file you saved before (e.g. on your Desktop). When you have selected the file, click **Next**.

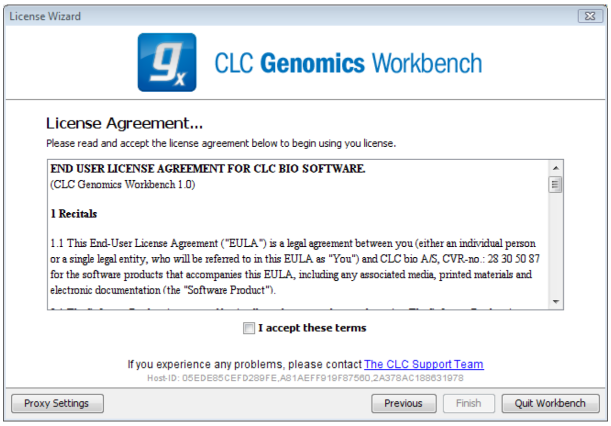
Accepting the license agreement

Regardless of which option you chose above, you will now see the dialog shown in figure 1.18.

Please read the License agreement carefully before clicking **I accept these terms** and **Finish**.

1.4.5 Configure license server connection

If you organization has installed a license server, you can use a floating license. The license server has a set of licenses that can be used on all computers on the network. If the server has



License Agreement...

Please read and accept the license agreement below to begin using you license.

END USER LICENSE AGREEMENT FOR CLC BIO SOFTWARE.
(CLC Genomics Workbench 1.0)

1 Recitals

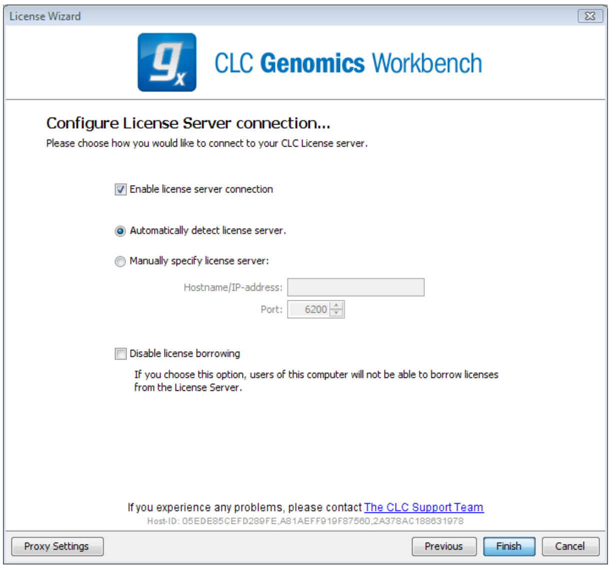
1.1 This End-User License Agreement ("EULA") is a legal agreement between you (either an individual person or a single legal entity, who will be referred to in this EULA as "You") and CLC bio A/S, CVR-no.: 28 30 50 87 for the software products that accompanies this EULA, including any associated media, printed materials and electronic documentation (the "Software Product").

□ **I accept these terms**

If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.18: *Read the license agreement carefully.*

e.g. 10 licenses, it means that maximum 10 computers can use a license *simultaneously*. When you have selected this option and click **Next**, you will see the dialog shown in figure 1.19.



Configure License Server connection...

Please choose how you would like to connect to your CLC License server.

☑ Enable license server connection

● Automatically detect license server.

○ Manually specify license server:

Hostname/IP-address:
Port: 6200

□ Disable license borrowing

If you choose this option, users of this computer will not be able to borrow licenses from the License Server.

If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.19: *Connecting to a license server.*

This dialog lets you specify how to connect to the license server:

- **Connect to a license server**. Check this option if you wish to use the license server.
- **Automatically detect license server**. By checking this option you do not have to enter more information to connect to the server.
- **Manually specify license server**. There can be technical limitations which mean that the license server cannot be detected automatically, and in this case you need to specify more options manually:
  - **Host name**. Enter the address for the licenser server.
  - **Port**. Specify which port to use.
- **Disable license borrowing on this computer**. If you do not want users of the computer to borrow a license (see section 1.4.5), you can check this option.

Borrow a license

A floating license can only be used when you are connected to the license server. If you wish to use the *CLC Genomics Workbench* when you are not connected to the server, you can *borrow* a license. Borrowing a license means that you take one of the floating licenses available on the server and borrow it for a specified amount of time. During this time period, there will be one less floating license available on the server.

At the point where you wish to borrow a license, you have to be connected to the license server. The procedure for borrowing is this:

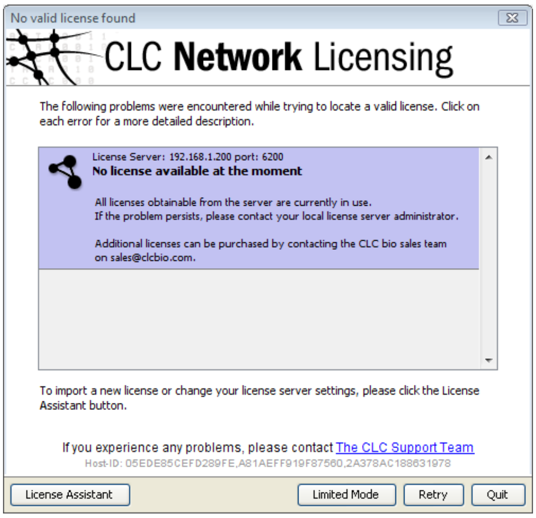
1. Click **Help** | **License Manager** to display the dialog shown in figure 1.22.
2. Use the checkboxes to select the license(s) that you wish to borrow.
3. Select how long time you wish to borrow the license, and click **Borrow Licenses**.
4. You can now go offline and work with *CLC Genomics Workbench*.
5. When the borrow time period has elapsed, you have to connect to the license server again to use *CLC Genomics Workbench*.
6. When the borrow time period has elapsed, the license server will make the floating license available for other users.

Note that the time period is not the period of time that you actually use the Workbench.

Note! When your organization's license server is installed, license borrowing can be turned off. In that case, you will not be able to borrow licenses.

No license available...

If all the licenses on the server are in use, you will see a dialog as shown in figure 1.20 when you start the Workbench.



The following problems were encountered while trying to locate a valid license. Click on each error for a more detailed description.

License Server: 192.168.1.200 port: 6200
No license available at the moment

All licenses obtainable from the server are currently in use.
If the problem persists, please contact your local license server administrator.

Additional licenses can be purchased by contacting the CLC bio sales team on sales@clcbio.com.

To import a new license or change your license server settings, please click the License Assistant button.

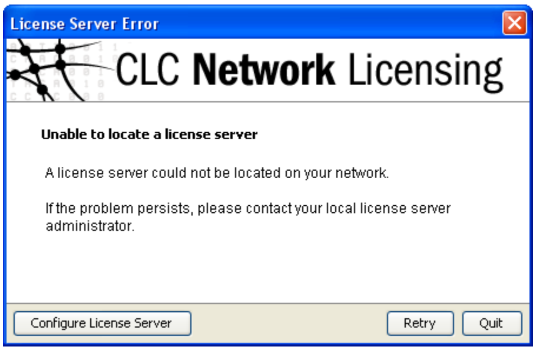
If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC188631978

Figure 1.20: *No more licenses available on the server.*

In this case, please contact your organization's license server administrator. To purchase additional licenses, contact sales@clcbio.com.

You can also click the **Limited Mode** button (see section 1.4.6).

If your connection to the license server is lost, you will see a dialog as shown in figure 1.21.



Unable to locate a license server

A license server could not be located on your network.

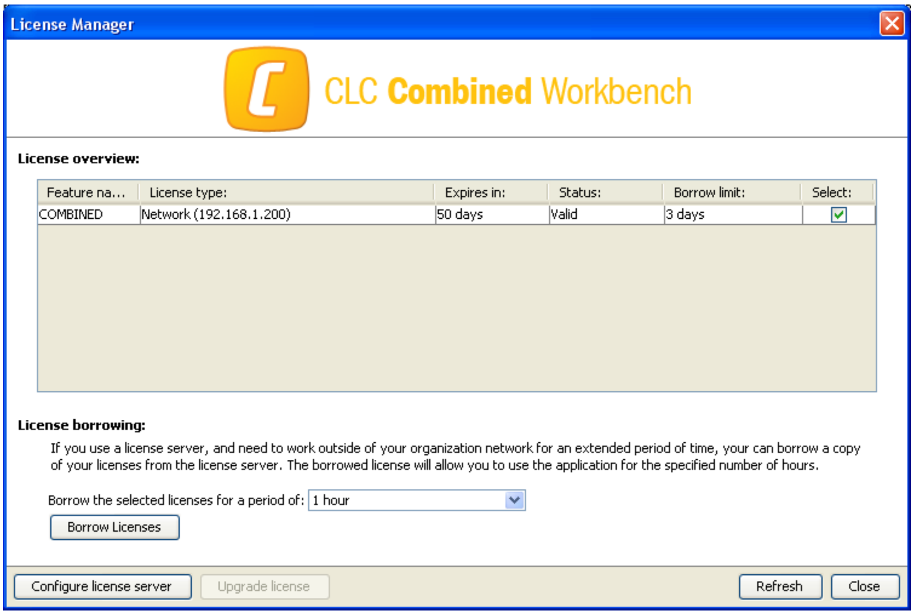
If the problem persists, please contact your local license server administrator.

Figure 1.21: *Unable to contact license server.*

In this case, you need to make sure that you have access to the license server, and that the server is running. However, there may be situations where you wish to use another license, or see information about the license you currently use. In this case, open the license manager:

Help | **License Manager** ( )

The license manager is shown in figure 1.22.



License overview:

Feature na...	License type:	Expires in:	Status:	Borrow limit:	Select:
COMBINED	Network (192.168.1.200)	50 days	Valid	3 days	☑

License borrowing:

If you use a license server, and need to work outside of your organization network for an extended period of time, your can borrow a copy of your licenses from the license server. The borrowed license will allow you to use the application for the specified number of hours.

Borrow the selected licenses for a period of: 1 hour

Figure 1.22: *The license manager.*

Besides letting you borrow licenses (see section 1.4.5), this dialog can be used to:

- See information about the license (e.g. what kind of license, when it expires)
- Configure how to connect to a license server (**Configure License Server** the button at the lower left corner). Clicking this button will display a dialog similar to figure 1.19.

- Upgrade from an evaluation license by clicking the **Upgrade license** button. This will display the dialog shown in figure 1.1.

If you wish to switch away from using a floating license, click **Configure License Server** and choose not to connect to a license server in the dialog. When you restart *CLC Genomics Workbench*, you will be asked for a license as described in section 1.4.

1.4.6 Limited mode

We have created the limited mode to prevent a situation where you are unable to access your data because you do not have a license. When you run in limited mode, a lot of the tools in the Workbench are not available, but you still have access to your data (also when stored in a *CLC Bioinformatics Database*). When running in limited mode, the functionality is equivalent to the *CLC Sequence Viewer* (see section A).

To get out of the limited mode and run the Workbench normally, restart the Workbench. When you restart the Workbench will try to find a proper license and if it does, it will start up normally. If it can't find a license, you will again have the option of running in limited mode.

1.5 About CLC Workbenches

In November 2005 CLC bio released two Workbenches: *CLC Free Workbench* and *CLC Protein Workbench*. *CLC Protein Workbench* is developed from the free version, giving it the well-tested user friendliness and look & feel. However, the *CLC Protein Workbench* includes a range of more advanced analyses.

In March 2006, *CLC DNA Workbench* (formerly *CLC Gene Workbench*) and *CLC Main Workbench* were added to the product portfolio of CLC bio. Like *CLC Protein Workbench*, *CLC DNA Workbench* builds on *CLC Free Workbench*. It shares some of the advanced product features of *CLC Protein Workbench*, and it has additional advanced features. *CLC Main Workbench* holds all basic and advanced features of the *CLC Workbenches*.

In June 2007, *CLC RNA Workbench* was released as a sister product of *CLC Protein Workbench* and *CLC DNA Workbench*. *CLC Main Workbench* now also includes all the features of *CLC RNA Workbench*.

In March 2008, the *CLC Free Workbench* changed name to *CLC Sequence Viewer*.

In June 2008, the first version of the *CLC Genomics Workbench* was released due to an extraordinary demand for software capable of handling sequencing data from the new high-throughput sequencing systems like 454, Illumina Genome Analyzer and SOLiD.

For an overview of which features all the applications include, see http://www.clcbio.com/features.

In December 2006, CLC bio released a **Software Developer Kit** which makes it possible for anybody with a knowledge of programming in Java to develop plug-ins. The plug-ins are fully integrated with the CLC Workbenches and the Viewer and provide an easy way to customize and extend their functionalities.

All our software will be improved continuously. If you are interested in receiving news about updates, you should register your e-mail and contact data on http://www.clcbio.com, if you haven't already registered when you downloaded the program.

1.5.1 New program feature request

The CLC team is continuously improving the *CLC Genomics Workbench* with our users' interests in mind. Therefore, we welcome all requests and feedback from users, and hope suggest new features or more general improvements to the program on support@clcbio.com.

1.5.2 Report program errors

CLC bio is doing everything possible to eliminate program errors. Nevertheless, some errors might have escaped our attention. If you discover an error in the program, you can use the **Report a Program Error** function in the **Help** menu of the program to report it. In the **Report a Program Error** dialog you are asked to write your e-mail address (optional). This is because we would like to be able to contact you for further information about the error or for helping you with the problem.

Note! No personal information is sent via the error report. Only the information which can be seen in the **Program Error Submission Dialog** is submitted.

You can also write an e-mail to support@clcbio.com. Remember to specify how the program error can be reproduced.

All errors will be treated seriously and with gratitude.

We appreciate your help.

Start in safe mode

If the program becomes unstable on start-up, you can start it in **Safe mode**. This is done by pressing and holding down the Shift button while the program starts.

When starting in safe mode, the user settings (e.g. the settings in the **Side Panel**) are deleted and cannot be restored. Your data stored in the **Navigation Area** is not deleted. When started in safe mode, some of the functionalities are missing, and you will have to restart the *CLC Genomics Workbench* again (without pressing Shift).

1.5.3 CLC Sequence Viewer vs. Workbenches

The advanced analyses of the commercial workbenches, *CLC Protein Workbench*, *CLC RNA Workbench* and *CLC DNA Workbench* are not present in *CLC Sequence Viewer*. Likewise, some advanced analyses are available in *CLC DNA Workbench* but not in *CLC RNA Workbench* or *CLC Protein Workbench*, and vice versa. All types of basic and advanced analyses are available in *CLC Main Workbench*.

However, the output of the commercial workbenches can be viewed in all other workbenches. This allows you to share the result of your advanced analyses from e.g. *CLC Main Workbench*, with people working with e.g. *CLC Sequence Viewer*. They will be able to view the results of your analyses, but not redo the analyses.

The CLC Workbenches and the CLC Sequence Viewer are developed for Windows, Mac and Linux platforms. Data can be exported/imported between the different platforms in the same easy way as when exporting/importing between two computers with e.g. Windows.

1.6 When the program is installed: Getting started

CLC Genomics Workbench includes an extensive **Help** function, which can be found in the **Help** menu of the program's **Menu bar**. The **Help** can also be shown by pressing F1. The help topics are sorted in a table of contents and the topics can be searched.

We also recommend our **Online presentations** where a product specialist from CLC bio demonstrates our software. This is a very easy way to get started using the program. Read more about online presentations here: http://clcbio.com/presentation.

1.6.1 Quick start

When the program opens for the first time, the background of the workspace is visible. In the background are three quick start shortcuts, which will help you getting started. These can be seen in figure 1.23.

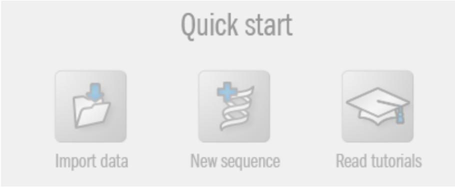


Figure 1.23: *Three available Quick start short cuts, available in the background of the workspace.*

The function of the three quick start shortcuts is explained here:

- **Import data.** Opens the **Import** dialog, which you let you browse for, and import data from your file system.
- **New sequence.** Opens a dialog which allows you to enter your own sequence.
- **Read tutorials.** Opens the tutorials menu with a number of tutorials. These are also available from the **Help** menu in the **Menu bar**.

1.6.2 Import of example data

It might be easier to understand the logic of the program by trying to do simple operations on existing data. Therefore *CLC Genomics Workbench* includes an example data set.

When downloading *CLC Genomics Workbench* you are asked if you would like to import the example data set. If you accept, the data is downloaded automatically and saved in the program. If you didn't download the data, or for some other reason need to download the data again, you have two options:

You can click **Install Example Data** () in the **Help** menu of the program. This installs the data automatically. You can also go to http://www.clcbio.com/download and download the example data from there.

If you download the file from the website, you need to import it into the program. See chapter 7.1 for more about importing data.

1.7 Plug-ins

When you install *CLC Genomics Workbench*, it has a standard set of features. However, you can upgrade and customize the program using a variety of plug-ins.

As the range of plug-ins is continuously updated and expanded, they will not be listed here. Instead we refer to `http://www.clcbio.com/plug-ins` for a full list of plug-ins with descriptions of their functionalities.

1.7.1 Installing plug-ins

Plug-ins are installed using the plug-in manager[2]:

Help in the Menu Bar | Plug-ins and Resources... ( )

or **Plug-ins ( ) in the Toolbar**

The plug-in manager has four tabs at the top:

- **Manage Plug-ins.** This is an overview of plug-ins that are installed.
- **Download Plug-ins.** This is an overview of available plug-ins on CLC bio's server.
- **Manage Resources.** This is an overview of resources that are installed.
- **Download Resources.** This is an overview of available resources on CLC bio's server.

To install a plug-in, click the **Download Plug-ins** tab. This will display an overview of the plug-ins that are available for download and installation (see figure 1.24).

Clicking a plug-in will display additional information at the right side of the dialog. This will also display a button: **Download and Install**.

Click the plug-in and press **Download and Install**. A dialog displaying progress is now shown, and the plug-in is downloaded and installed.

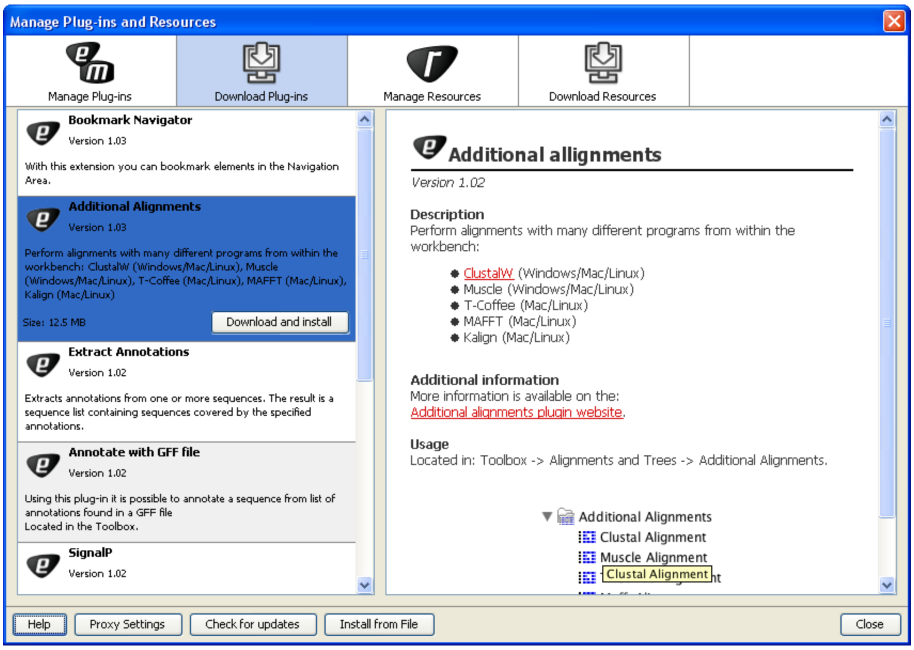
If the plug-in is not shown on the server, and you have it on your computer (e.g. if you have downloaded it from our web-site), you can install it by clicking the **Install from File** button at the bottom of the dialog. This will open a dialog where you can browse for the plug-in. The plug-in file should be a file of the type ".cpa".

When you close the dialog, you will be asked whether you wish to restart the *CLC Genomics Workbench*. The plug-in will not be ready for use before you have restarted.

1.7.2 Uninstalling plug-ins

Plug-ins are uninstalled using the plug-in manager:

[2] In order to install plug-ins on Windows Vista, the Workbench must be run in administrator mode: Right-click the program shortcut and choose "Run as Administrator". Then follow the procedure described below. When you start the Workbench after installing the plug-in, it should also be run in administrator mode.



Bookmark Navigator
Version 1.03
With this extension you can bookmark elements in the Navigation Area.

Additional Alignments
Version 1.03
Perform alignments with many different programs from within the workbench: ClustalW (Windows/Mac/Linux), Muscle (Windows/Mac/Linux), T-Coffee (Mac/Linux), MAFFT (Mac/Linux), Kalign (Mac/Linux)
Size: 12.5 MB

Extract Annotations
Version 1.02
Extracts annotations from one or more sequences. The result is a sequence list containing sequences covered by the specified annotations.

Annotate with GFF file
Version 1.02
Using this plug-in it is possible to annotate a sequence from list of annotations found in a GFF file
Located in the Toolbox.

SignalP
Version 1.02

Additional allignments
Version 1.02

Description
Perform alignments with many different programs from within the workbench:

- ClustalW (Windows/Mac/Linux)
- Muscle (Windows/Mac/Linux)
- T-Coffee (Mac/Linux)
- MAFFT (Mac/Linux)
- Kalign (Mac/Linux)

Additional information
More information is available on the:
Additional alignments plugin website.

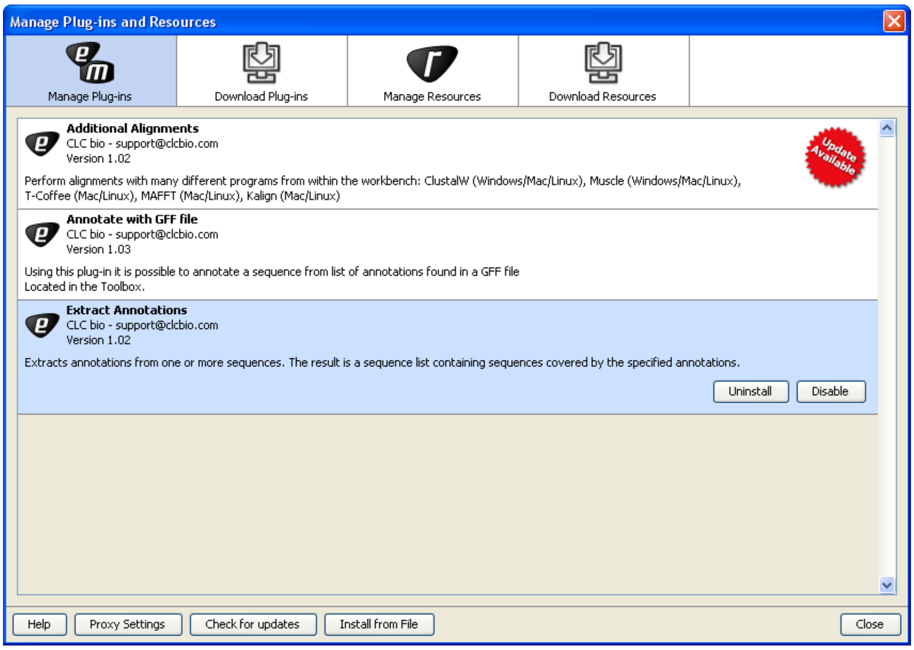
Usage
Located in: Toolbox -> Alignments and Trees -> Additional Alignments.

Figure 1.24: *The plug-ins that are available for download.*

Help in the Menu Bar | Plug-ins and Resources... (⎙**)**

or **Plug-ins (**⎙**) in the Toolbar**

This will open the dialog shown in figure 1.25.



Additional Alignments
CLC bio - support@clcbio.com
Version 1.02
Perform alignments with many different programs from within the workbench: ClustalW (Windows/Mac/Linux), Muscle (Windows/Mac/Linux), T-Coffee (Mac/Linux), MAFFT (Mac/Linux), Kalign (Mac/Linux)

Annotate with GFF file
CLC bio - support@clcbio.com
Version 1.03
Using this plug-in it is possible to annotate a sequence from list of annotations found in a GFF file
Located in the Toolbox.

Extract Annotations
CLC bio - support@clcbio.com
Version 1.02
Extracts annotations from one or more sequences. The result is a sequence list containing sequences covered by the specified annotations.

Figure 1.25: *The plug-in manager with plug-ins installed.*

The installed plug-ins are shown in this dialog. To uninstall:

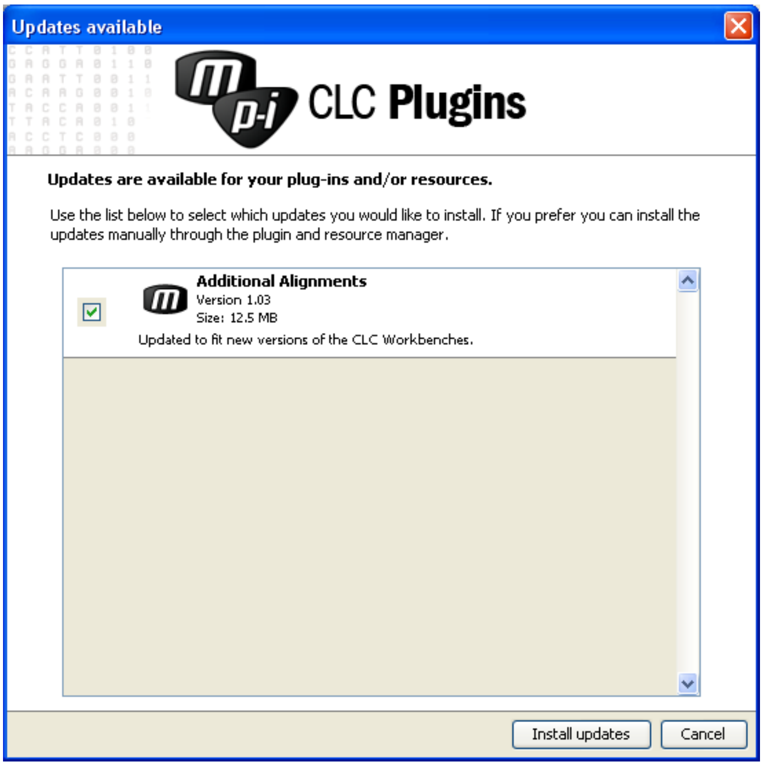
Click the plug-in | Uninstall

If you do not wish to completely uninstall the plug-in but you don't want it to be used next time you start the Workbench, click the **Disable** button.

When you close the dialog, you will be asked whether you wish to restart the workbench. The plug-in will not be uninstalled before the workbench is restarted.

1.7.3 Updating plug-ins

If a new version of a plug-in is available, you will get a notification during start-up as shown in figure 1.26.



Updates are available for your plug-ins and/or resources.

Use the list below to select which updates you would like to install. If you prefer you can install the updates manually through the plugin and resource manager.

☑ **Additional Alignments**
Version 1.03
Size: 12.5 MB
Updated to fit new versions of the CLC Workbenches.

Figure 1.26: *Plug-in updates.*

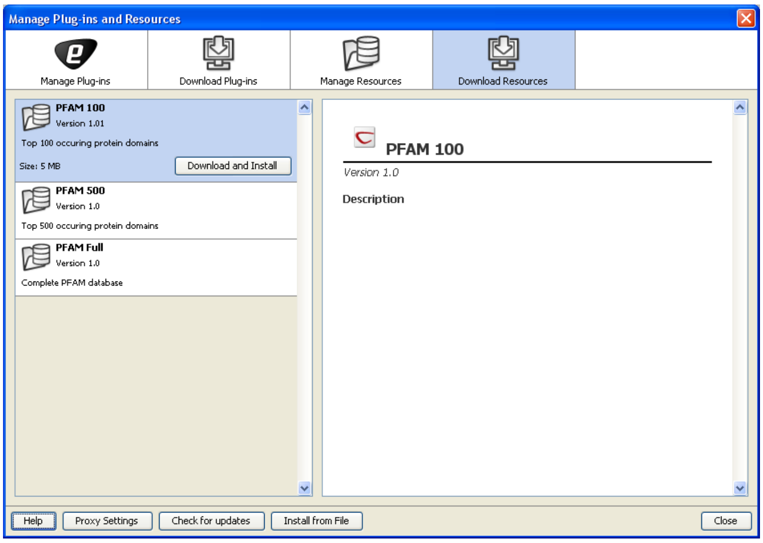
In this list, select which plug-ins you wish to update, and click **Install Updates**. If you press **Cancel** you will be able to install the plug-ins later by clicking **Check for Updates** in the Plug-in manager (see figure 1.25).

1.7.4 Resources

Resources are downloaded, installed, un-installed and updated the same way as plug-ins. Click the **Download Resources** tab at the top of the plug-in manager, and you will see a list of available resources (see figure 1.27).

Currently, the only resources available are PFAM databases (for use with *CLC Protein Workbench* and *CLC Main Workbench*).

Because procedures for downloading, installation, uninstallation and updating are the same as for plug-ins see section 1.7.1 and section 1.7.2 for more information.



PFAM 100
Version 1.01
Top 100 occuring protein domains
Size: 5 MB

PFAM 500
Version 1.0
Top 500 occuring protein domains

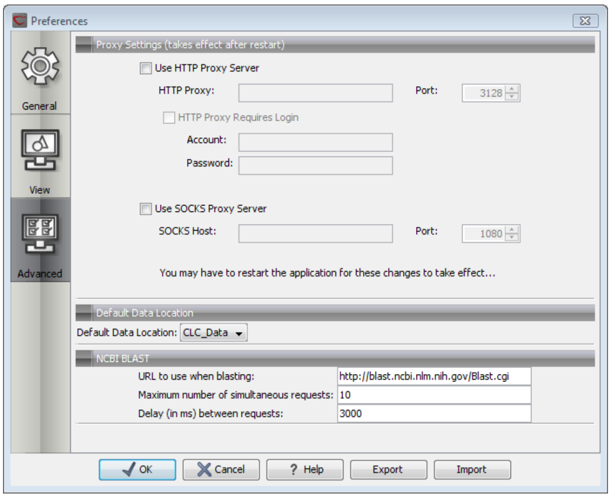
PFAM Full
Version 1.0
Complete PFAM database

PFAM 100
Version 1.0
Description

Figure 1.27: *Resources available for download.*

1.8 Network configuration

If you use a proxy server to access the Internet you must configure *CLC Genomics Workbench* to use this. Otherwise you will not be able to perform any online activities (e.g. searching GenBank). *CLC Genomics Workbench* supports the use of a HTTP-proxy and an anonymous SOCKS-proxy.



□ Use HTTP Proxy Server
HTTP Proxy: Port: 3128
□ HTTP Proxy Requires Login
Account:
Password:

□ Use SOCKS Proxy Server
SOCKS Host: Port: 1080

You may have to restart the application for these changes to take effect...

Default Data Location: CLC_Data

URL to use when blasting: http://blast.ncbi.nlm.nih.gov/Blast.cgi
Maximum number of simultaneous requests: 10
Delay (in ms) between requests: 3000

Figure 1.28: *Adjusting proxy preferences.*

To configure your proxy settings, open *CLC Genomics Workbench*, and go to the **Advanced**-tab of the **Preferences** dialog (figure 1.28) and enter the appropriate information. The **Preferences** dialog is opened from the **Edit** menu.

You have the choice between a HTTP-proxy and a SOCKS-proxy. *CLC Genomics Workbench* only supports the use of a SOCKS-proxy that does not require authorization.

If you have any problems with these settings you should contact your systems administrator.

1.9 The format of the user manual

This user manual offers support to Windows, Mac OS X and Linux users. The software is very similar on these operating systems. In areas where differences exist, these will be described separately. However, the term "right-click" is used throughout the manual, but some Mac users may have to use Ctrl+click in order to perform a "right-click" (if they have a single-button mouse).

The most recent version of the user manuals can be downloaded from `http://www.clcbio.com/usermanuals`.

The user manual consists of four parts.

- The **first part** includes the introduction and some tutorials showing how to apply the most significant functionalities of *CLC Genomics Workbench*.
- The **second part** describes in detail how to operate all the program's basic functionalities.
- The **third part** digs deeper into some of the bioinformatic features of the program. In this part, you will also find our "Bioinformatics explained" sections. These sections elaborate on the algorithms and analyses of *CLC Genomics Workbench* and provide more general knowledge of bioinformatic concepts.
- The **fourth part** is the Appendix and Index.

Each chapter includes a short table of contents.

1.9.1 Text formats

In order to produce a clearly laid-out content in this manual, different formats are applied:

- A feature in the program is in bold starting with capital letters. (Example: **Navigation Area**)
- An explanation of how a particular function is activated, is illustrated by "|" and bold. (E.g.: **select the element** | **Edit** | **Rename**)

Chapter 2

Tutorials

Contents

This chapter contains tutorials representing some of the features of *CLC Genomics Workbench*. The first tutorials are meant as a short introduction to operating the program. The last tutorials give examples of how to use some of the main features of *CLC Genomics Workbench*. **Watch video tutorials at `http://www.clcbio.com/tutorials`.**

2.1 Tutorial: Getting started

This brief tutorial will take you through the most basic steps of working with *CLC Genomics Workbench*. The tutorial introduces the user interface, shows how to create a folder, and demonstrates how to import your own existing data into the program.

When you open *CLC Genomics Workbench* for the first time, the user interface looks like figure 2.1.

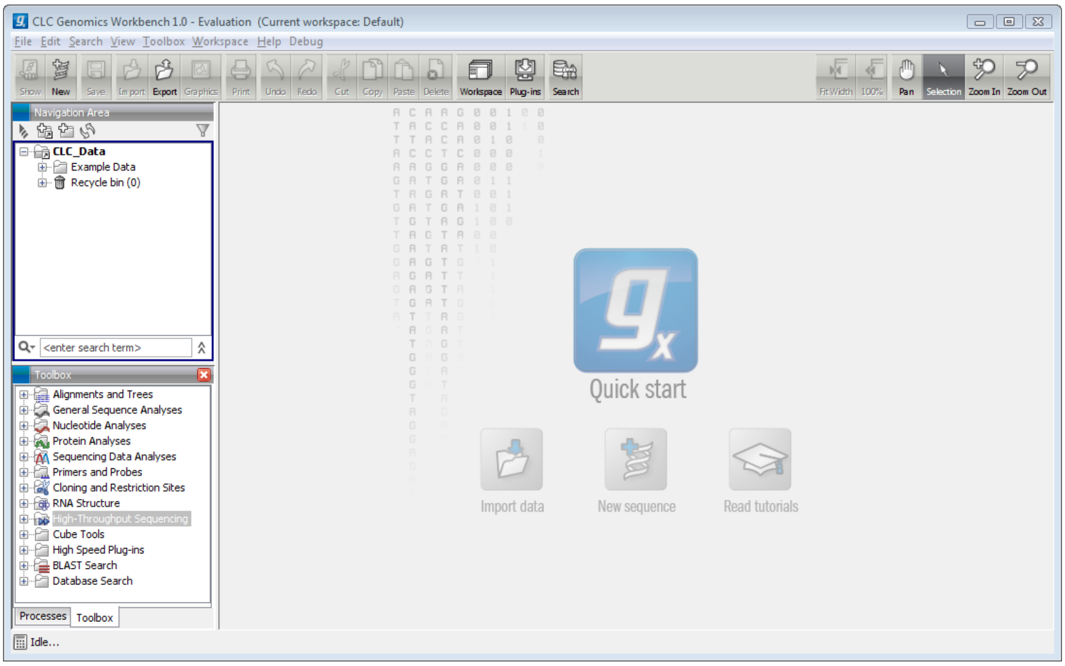

CLC Genomics Workbench 1.0 - Evaluation (Current workspace: Default)
File Edit Search View Toolbox Workspace Help Debug
Show New Save Import Export Graphics Print Undo Redo Cut Copy Paste Delete Workspace Plug-ins Search
Fit Width 100% Pan Selection Zoom In Zoom Out
Navigation Area
CLC_Data
Example Data
Recycle bin (0)
<enter search term>
Toolbox
Alignments and Trees
General Sequence Analyses
Nucleotide Analyses
Protein Analyses
Sequencing Data Analyses
Primers and Probes
Cloning and Restriction Sites
RNA Structure
High-Throughput Sequencing
Cube Tools
High Speed Plug-ins
BLAST Search
Database Search
Processes Toolbox
Idle...
Quick start
Import data
New sequence
Read tutorials

Figure 2.1: *The user interface as it looks when you start the program for the first time. (Windows version of* **CLC Genomics Workbench***. The interface is similar for Mac and Linux.)*

At this stage, the important issues are the **Navigation Area** and the **View Area**.

The **Navigation Area** to the left is where you keep all your data for use in the program. Most analyses of *CLC Genomics Workbench* require that the data is saved in the **Navigation Area**. There are several ways to get data into the **Navigation Area**, and this tutorial describes how to import existing data.

The **View Area** is the main area to the right. This is where the data can be 'viewed'. In general, a **View** is a display of a piece of data, and the **View Area** can include several **Views**. The **Views** are represented by tabs, and can be organized e.g. by using 'drag and drop'.

2.1.1 Creating a a folder

When *CLC Genomics Workbench* is started there is one element in the **Navigation Area** called **CLC_Data**[1]. This element is a **Location**. A location points to a folder on your computer where your data for use with *CLC Genomics Workbench* is stored.

The data in the location can be organized into folders. Create a folder:

File | **New** | **Folder ( )**

[1] If you have downloaded the example data, this will be placed as a folder in *CLC_Data*

or **Ctrl + Shift + N (⌘ + Shift + N on Mac)**

Name the folder 'My folder' and press **Enter**.

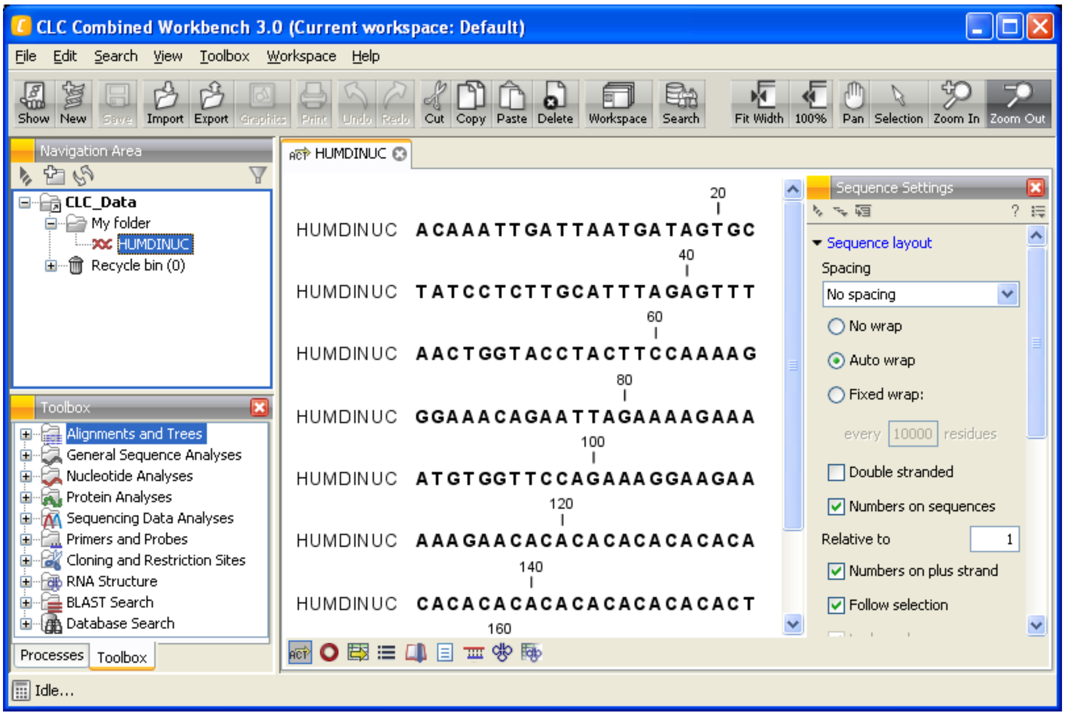
2.1.2 Import data

Next, we want to import a sequence called HUMDINUC.fsa (FASTA format) from our own Desktop into the new 'My folder'. (This file is chosen for demonstration purposes only - you may have another file on your desktop, which you can use to follow this tutorial. You can import all kinds of files.)

In order to import the HUMDINUC.fsa file:

Select 'My folder' | Import (📂) in the Toolbar | navigate to HUMDINUC.fsa on the desktop | Select

The sequence is imported into the folder that was selected in the **Navigation Area**, before you clicked **Import**. Double-click the sequence in the **Navigation Area** to view it. The final result looks like figure 2.2.


CLC Combined Workbench 3.0 (Current workspace: Default)
File Edit Search View Toolbox Workspace Help
Show New Save Import Export Graphics Print Undo Redo Cut Copy Paste Delete Workspace Search Fit Width 100% Pan Selection Zoom In Zoom Out
Navigation Area
CLC_Data
My folder
HUMDINUC
Recycle bin (0)
Toolbox
Alignments and Trees
General Sequence Analyses
Nucleotide Analyses
Protein Analyses
Sequencing Data Analyses
Primers and Probes
Cloning and Restriction Sites
RNA Structure
BLAST Search
Database Search
Processes Toolbox
HUMDINUC
Sequence Settings
Sequence layout
Idle...

```
                                 20
HUMDINUC  ACAAATTGATTAATGATAGTGC
                               40
HUMDINUC  TATCCTCTTGCATTTAGAGTTT
                             60
HUMDINUC  AACTGGTACCTACTTCCAAAAG
                           80
HUMDINUC  GGAAACAGAATTAGAAAAGAAA
                         100
HUMDINUC  ATGTGGTTCCAGAAAGGAAGAA
                       120
HUMDINUC  AAAGAACACACACACACACACA
                     140
HUMDINUC  CACACACACACACACACACACT
                   160
```

Spacing: No spacing

○ No wrap
● Auto wrap
○ Fixed wrap:
every 10000 residues

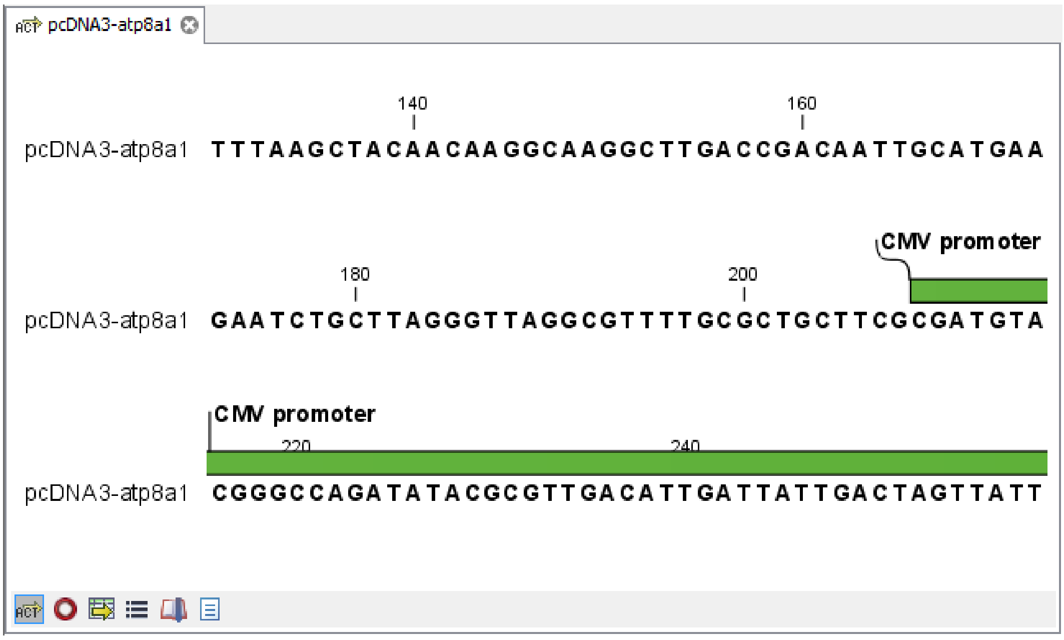
□ Double stranded
☑ Numbers on sequences
Relative to 1
☑ Numbers on plus strand
☑ Follow selection

Figure 2.2: *The HUMDINUC file is imported and opened.*

2.2 Tutorial: View sequence

This brief tutorial will take you through some different ways to display a sequence in the program. The tutorial introduces zooming on a sequence, dragging tabs, and opening selection in new view.

We will be working with the sequence called *pcDNA3-atp8a1* located in the 'Cloning' folder in the Example data. Double-click the sequence in the **Navigation Area** to open it. The sequence is displayed with annotations above it. (See figure 2.3).



```
pcDNA3-atp8a1  TTTAAGCTACAACAAGGCAAGGCTTGACCGACAATTGCATGAA
pcDNA3-atp8a1  GAATCTGCTTAGGGTTAGGCGTTTTGCGCTGCTTCGCGATGTA
pcDNA3-atp8a1  CGGGCCAGATATACGCGTTGACATTGATTATTGACTAGTTATT
```

Figure 2.3: *Sequence pcDNA3-atp8a1 opened in a view.*

As default, *CLC Genomics Workbench* displays a sequence with annotations (colored arrows on the sequence like the green promoter region annotation in figure 2.3) and zoomed to see the residues.

In this tutorial we want to have an overview of the whole sequence. Hence;

> **click Zoom Out (🔍) in the Toolbar | click the sequence until you can see the whole sequence**

This sequence is circular, which is indicated by << and >> at the beginning and the end of the sequence.

In the following we will show how the same sequence can be displayed in two different views - one linear view and one circular view. First, zoom in to see the residues again by using the **Zoom In** (🔍) or the **100%** ( ). Then we make a split view by:

> **press and hold the Ctrl-button on the keyboard (⌘ on Mac) | click Show as Circular (○) at the bottom of the view**

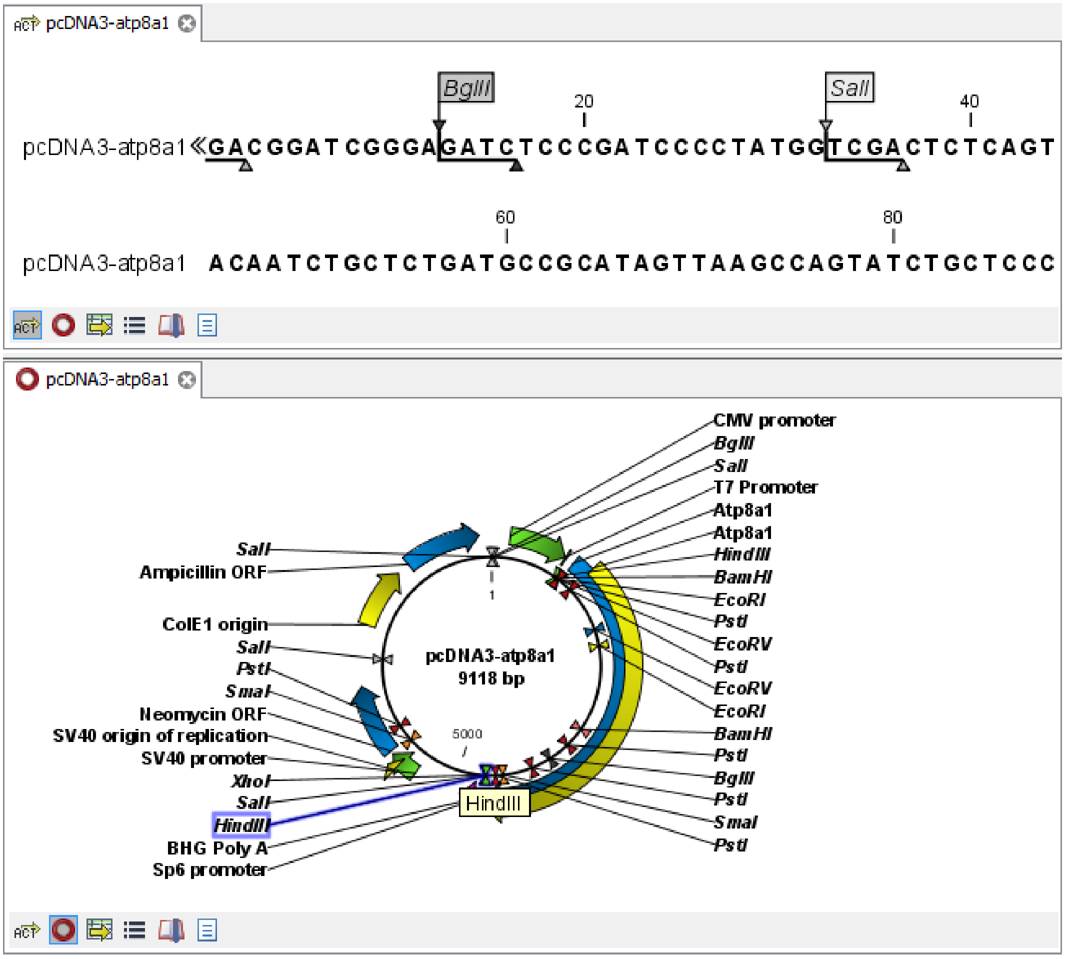
This opens an additional view of the vector with a circular display, as can be seen in figure 2.4.

Make a selection on the circular sequence (remember to switch to the **Selection** ( ) tool in the tool bar) and note that this selection is also reflected in the linear view above.

2.3 Tutorial: Side Panel Settings

This brief tutorial will show you how to use the **Side Panel** to change the way your sequences, alignments and other data are shown. You will also see how to save the changes that you made in the **Side Panel** .

Open the protein alignment located under *Protein orthologs* in the **Example data**. The initial view of the alignment has colored the residues according to the Rasmol color scheme, and the



```
                                                 20                                      40
pcDNA3-atp8a1 «GACGGATCGGGAGATCTCCCGATCCCCTATGGTCGACTCTCAGT

                                            60                                      80
pcDNA3-atp8a1  ACAATCTGCTCTGATGCCGCATAGTTAAGCCAGTATCTGCTCCC
```

pcDNA3-atp8a1
9118 bp

Figure 2.4: *The resulting two views which are split horizontally.*

alignment is automatically wrapped to fit the width of the view (shown in figure 2.5).

Now, we are going to modify how this alignment is displayed. For this, we use the settings in the **Side Panel** to the right. All the settings are organized into groups, which can be expanded / collapsed by clicking the name of the group. The first group is **Sequence Layout** which is expanded by default.

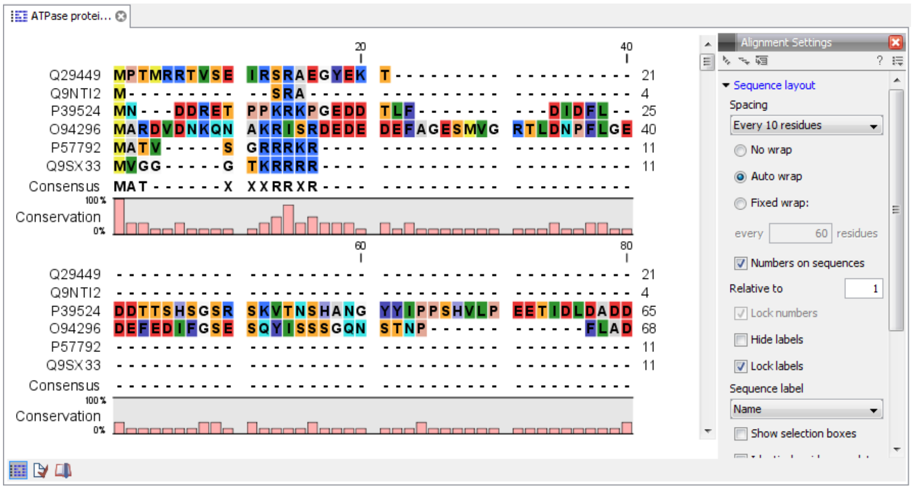
First, select **No wrap** in the **Sequence Layout**. This means that each sequence in the alignment is kept on the same line. To see more of the alignment, you now have to scroll horizontally.

Next, expand the **Annotation Layout** group and select **Show Annotations**. Set the **Offset** to "More offset" and set the **Label** to "Stacked".

Expand the **Annotation Types** group. Here you will see a list of the types annotation that are carried by the sequences in the alignment (see figure 2.6).

Check the "Region" annotation type, and you will see the regions as red annotations on the sequences.

Next, we will change the way the residues are colored. Click the **Alignment Info** group and under **Conservation**, check "Background color". This will use a gradient as background color for the residues. You can adjust the coloring by dragging the small arrows above the color box.



```
                                              20                                  40
                                              |                                   |
Q29449       MPTMRRTVSE  IRSRAEGYEK  T---------  ----------  21
Q9NTI2       M---------  --SRA-----  ----------  ----------  4
P39524       MN---DDRET  PPKRKPGEDD  TLF-------  ---DIDFL--  25
O94296       MARDVDNKQN  AKRISRDEDE  DEFAGESMVG  RTLDNPFLGE  40
P57792       MATV-----S  GRRRKR----  ----------  ----------  11
Q9SX33       MVGG-----G  TKRRRR----  ----------  ----------  11
Consensus    MAT------X  XXRRXR----  ----------  ----------
        100%
Conservation
          0%
                                              60                                  80
                                              |                                   |
Q29449       ----------  ----------  ----------  ----------  21
Q9NTI2       ----------  ----------  ----------  ----------  4
P39524       DDTTSHSGSR  SKVTNSHANG  YYIPPSHVLP  EETIDLDADD  65
O94296       DEFEDIFGSE  SQYISSSGQN  STNP------  ------FLAD  68
P57792       ----------  ----------  ----------  ----------  11
Q9SX33       ----------  ----------  ----------  ----------  11
Consensus    ----------  ----------  ----------  ----------
        100%
Conservation
          0%
```

Sequence layout

Spacing: Every 10 residues

○ No wrap

● Auto wrap

○ Fixed wrap:

every 60 residues

☑ Numbers on sequences

Relative to 1

☑ Lock numbers

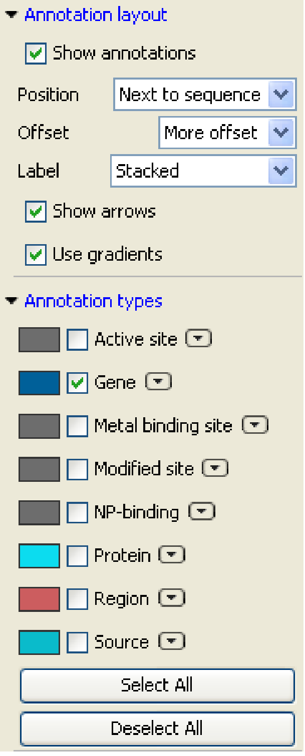
☐ Hide labels

☑ Lock labels

Sequence label: Name

☐ Show selection boxes

Figure 2.5: *The protein alignment as it looks when you open it with background color according to the Rasmol color scheme and automatically wrapped.*



Annotation layout

☑ Show annotations

Position: Next to sequence

Offset: More offset

Label: Stacked

☑ Show arrows

☑ Use gradients

Annotation types

☐ Active site

☑ Gene

☐ Metal binding site

☐ Modified site

☐ NP-binding

☐ Protein

☐ Region

☐ Source

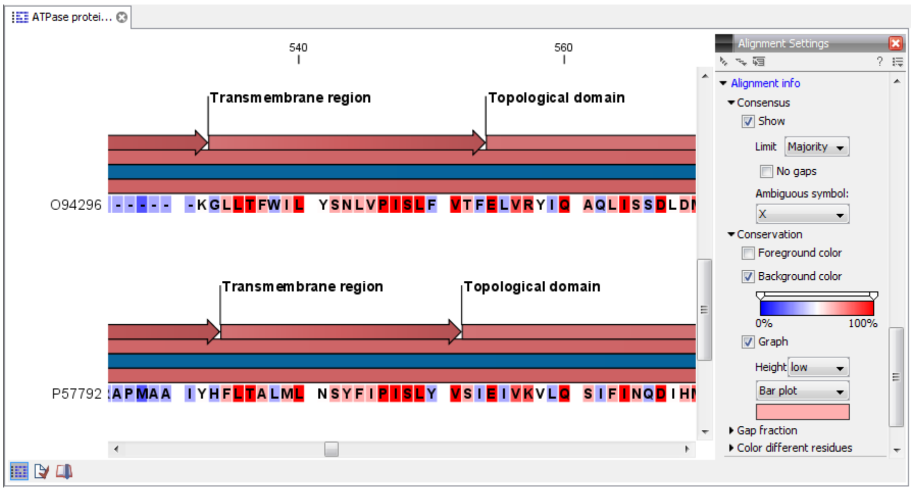
Figure 2.6: *The Annotation Layout and the Annotation Types in the Side Panel.*

2.3.1 Saving the settings in the Side Panel

Now the alignment should look similar to figure 2.7.

At this point, if you just close the view, the changes made to the **Side Panel** will not be saved. This means that you would have to perform the changes again next time you open the alignment. To save the changes to the **Side Panel**, click the **Save/Restore Settings** button (☰) at the top of the **Side Panel** and click **Save Settings** (see figure 2.8).

This will open the dialog shown in figure 2.9.



Alignment info

Consensus

☑ Show

Limit: Majority

□ No gaps

Ambiguous symbol: X

Conservation

□ Foreground color

☑ Background color

0% 100%

☑ Graph

Height: low

Bar plot

Gap fraction

Color different residues

Figure 2.7: *The alignment when all the above settings have been changed.*

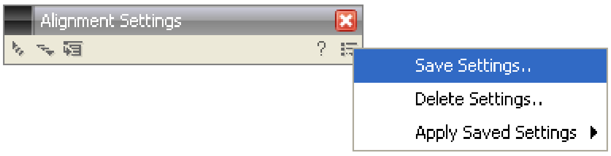
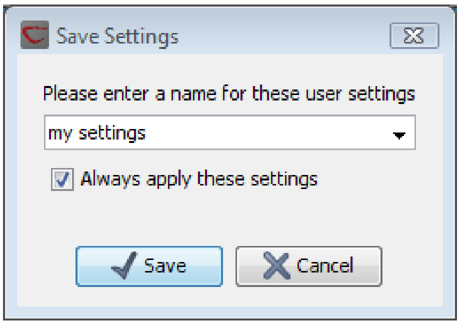


Figure 2.8: *Saving the settings of the Side Panel.*



Please enter a name for these user settings

my settings

☑ Always apply these settings

Figure 2.9: *Dialog for saving the settings of the Side Panel.*

In this way you can save the current state of the settings in the **Side Panel** so that you can apply them to alignments later on. If you check **Always apply these settings**, these settings will be applied every time you open a view of the alignment.

Type "My settings" in the dialog and click **Save**.

2.3.2 Applying saved settings

When you click the **Save/Restore Settings** button (☰) again and select **Apply Saved Settings**, you will see "My settings" in the menu together with some pre-defined settings that the *CLC Genomics Workbench* has created for you (see figure 2.10).

Whenever you open an alignment, you will be able to apply these settings. Each kind of view has its own list of settings that can be applied.

At the bottom of the list you will see the "CLC Standard Settings" which are the default settings for the view.

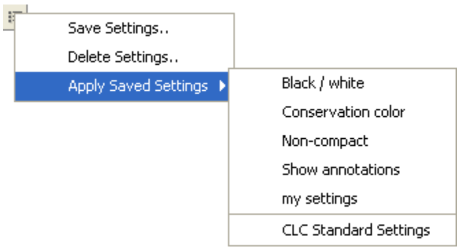


Figure 2.10: *Menu for applying saved settings.*

2.4 Tutorial: GenBank search and download

The *CLC Genomics Workbench* allows you to search the NCBI GenBank database directly from the program, giving you the opportunity to both open, view, analyze and save the search results without using any other applications. To conduct a search in NCBI GenBank from *CLC Genomics Workbench* you must be connected to the Internet.

This tutorial shows how to find a complete human hemoglobin DNA sequence in a situation where you do not know the accession number of the sequence.

To start the search:

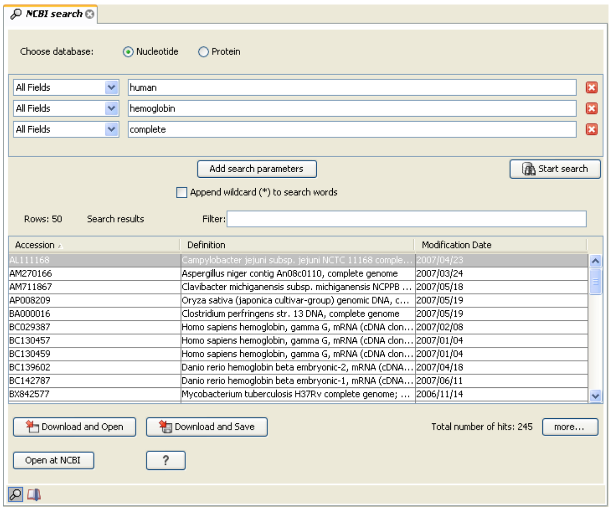
> **Search | Search for Sequences at NCBI ( )**

This opens the search view. We are searching for a DNA sequence, hence:

> **Nucleotide**

Now we are going to adjust parameters for the search. By clicking **Add search parameters** you activate an additional set of fields where you can enter search criteria. Each search criterion consists of a drop down menu and a text field. In the drop down menu you choose which part of the NCBI database to search, and in the text field you enter what to search for:

> **Click Add search parameters until three search criteria are available | choose Organism in the first drop down menu | write 'human' in the adjoining text field | choose All Fields in the second drop down menu | write 'hemoglobin' in the adjoining text field | choose All Fields in the third drop down menu | write 'complete' in the adjoining text field**



Choose database: ● Nucleotide ○ Protein

All Fields: human
All Fields: hemoglobin
All Fields: complete

□ Append wildcard (*) to search words

Rows: 50 Search results Filter:

Accession	Definition	Modification Date
AL111168	Campylobacter jejuni subsp. jejuni NCTC 11168 comple...	2007/04/23
AM270166	Aspergillus niger contig An08c0110, complete genome	2007/03/24
AM711867	Clavibacter michiganensis subsp. michiganensis NCPPB ...	2007/05/18
AP008209	Oryza sativa (japonica cultivar-group) genomic DNA, c...	2007/05/19
BA000016	Clostridium perfringens str. 13 DNA, complete genome	2007/05/19
BC029387	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/02/08
BC130457	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/01/04
BC130459	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/01/04
BC139602	Danio rerio hemoglobin beta embryonic-2, mRNA (cDNA...	2007/04/18
BC142787	Danio rerio hemoglobin beta embryonic-1, mRNA (cDNA...	2007/06/11
BX842577	Mycobacterium tuberculosis H37Rv complete genome; ...	2006/11/14

Total number of hits: 245

Figure 2.11: *NCBI search view.*

Click **Start search ()** to commence the search in NCBI.

2.4.1 Searching for matching objects

When the search is complete, the list of hits is shown. If the desired complete human hemoglobin DNA sequence is found, the sequence can be viewed by double-clicking it in the list of hits from the search. If the desired sequence is not shown, you can click the 'More' button below the list to see more hits.

2.4.2 Saving the sequence

The sequences which are found during the search can be displayed by double-clicking in the list of hits. However, this does not save the sequence. You can save one or more sequence by selecting them and:

click Download and Save

or **drag the sequences into the Navigation Area**

2.5 Tutorial: De novo assembly and BLAST

This tutorial takes you through some of the tools for a typical *de novo* sequencing work flow with a data set from a high-throughput sequencing machine. As an example we use an *E. coli* data set consisting of a little more than 400,000 reads from a 454 sequencer. [2]

2.5.1 Importing the data

First, download the data set from our web site: `http://download.clcbio.com/testdata/raw_data/454.zip`. Unzip the file somewhere on your computer (e.g. the Desktop).

Start the *CLC Genomics Workbench* and import the data:

File | Import High-throughput Sequencing | Roche 454

This will bring up the dialog shown in figure 2.12

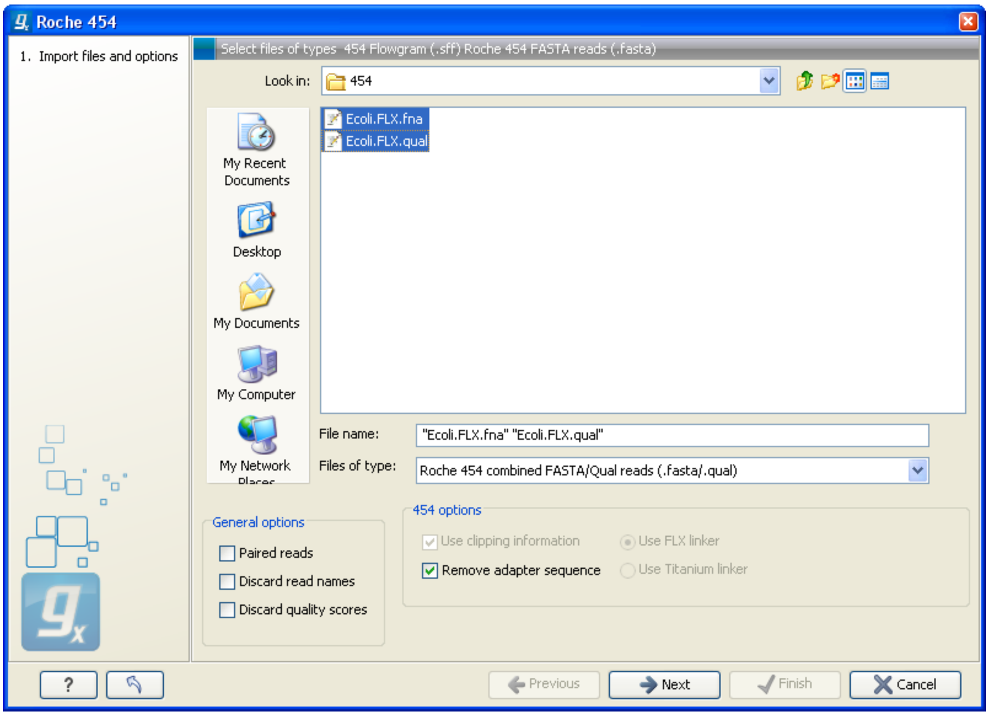
Select the `Ecoli.FLX.fna` and `Ecoli.FLX.qual` files that come from the downloaded zip file. Make sure the **Remove adapter sequence** checkbox is checked and that the **Paired reads** checkbox is NOT checked. The option to discard read names is not significant in this context because of the relatively small amount of reads. Click **Next**, choose to **Save** and click **Finish**.

After a short while, the reads will be imported.

2.5.2 Assembly

The first step is to do a *de novo* assembly of the reads. Briefly explained, this means that we want the *CLC Genomics Workbench* to create longer contiguous sequences out of the relatively short reads that are on average 235 bp long:

[2] Note that there are special system requirements for *CLC Genomics Workbench*, see `http://www.clcbio.com/index.php?id=72`



1. Import files and options

Select files of types 454 Flowgram (.sff) Roche 454 FASTA reads (.fasta)

Look in: 454

Ecoli.FLX.fna
Ecoli.FLX.qual

File name: "Ecoli.FLX.fna" "Ecoli.FLX.qual"

Files of type: Roche 454 combined FASTA/Qual reads (.fasta/.qual)

General options

- □ Paired reads
- □ Discard read names
- □ Discard quality scores

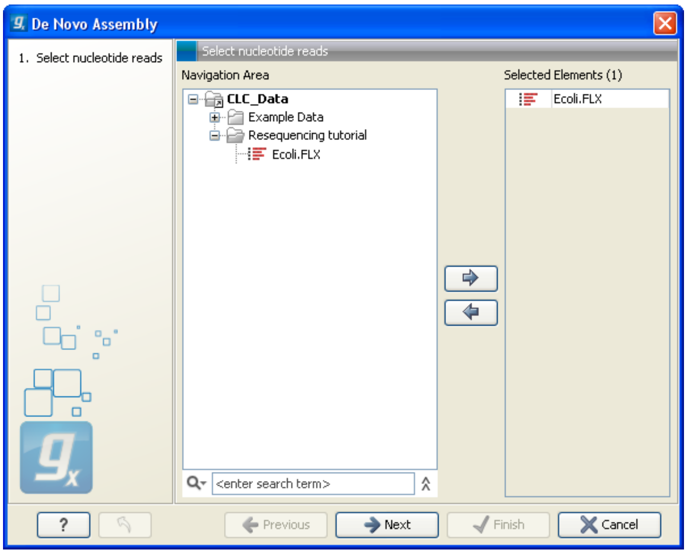
454 options

- ☑ Use clipping information
- ☑ Remove adapter sequence
- ● Use FLX linker
- ○ Use Titanium linker

Figure 2.12: *Choosing the file you wish to import.*

Toolbox | High-throughput Sequencing ( ) | De Novo Assembly ( )

This shows the dialog in figure 2.13.



1. Select nucleotide reads

Select nucleotide reads

Navigation Area

- CLC_Data
  - Example Data
  - Resequencing tutorial
    - Ecoli.FLX

<enter search term>

Selected Elements (1)

Ecoli.FLX

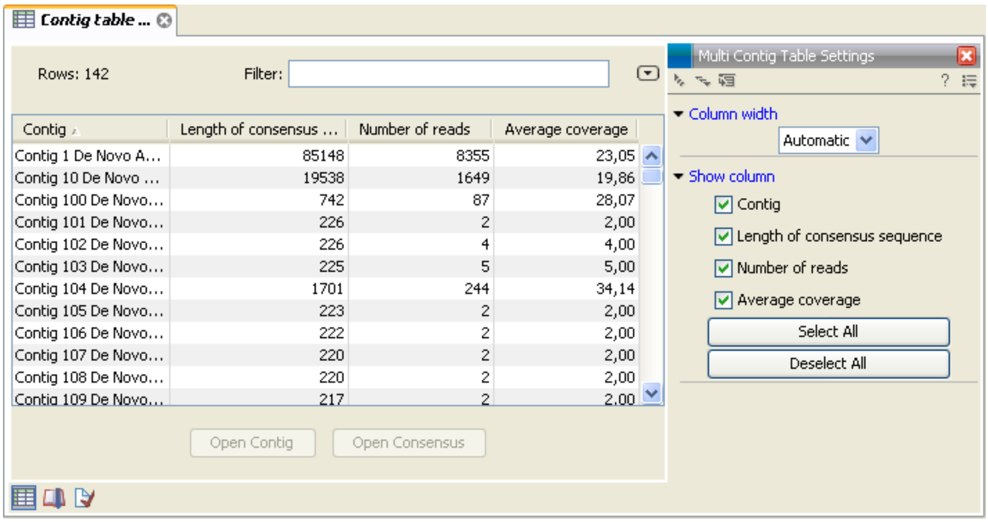
Figure 2.13: *Select sequence list containing the reads.*

Select the `Ecoli.FLX` ( ) sequence list and add it to the panel to the right. Click **Next**, **Next** and **Finish**.

For information about the settings, click the **Help** (?) button.

The assembly will take several minutes. There are two steps in the assembly process. In the first step, contig sequences are created by assembling all the reads. Second, all the reads are mapped back using the contig sequences as references. This is done to create an alignment of all the reads (if you only wish to see the contiguous sequences without the reads, you should choose **Create simple contig sequence** before you clicked Finish).

The result of the assembly is a mapping table as shown in figure 2.14.



Rows: 142 Filter:

Contig	Length of consensus ...	Number of reads	Average coverage
Contig 1 De Novo A...	85148	8355	23,05
Contig 10 De Novo ...	19538	1649	19,86
Contig 100 De Novo...	742	87	28,07
Contig 101 De Novo...	226	2	2,00
Contig 102 De Novo...	226	4	4,00
Contig 103 De Novo...	225	5	5,00
Contig 104 De Novo...	1701	244	34,14
Contig 105 De Novo...	223	2	2,00
Contig 106 De Novo...	222	2	2,00
Contig 107 De Novo...	220	2	2,00
Contig 108 De Novo...	220	2	2,00
Contig 109 De Novo...	217	2	2,00

Column width: Automatic

Show column
☑ Contig
☑ Length of consensus sequence
☑ Number of reads
☑ Average coverage

Figure 2.14: *A mapping table.*

Each row in the table represents a contig. There should be around 142 contigs (the number can vary a little from version to version). Double-clicking a row will open the contig with the mapped reads.

2.5.3 BLAST the results

One way to investigate the contigs would be to BLAST them against public databases. In this tutorial we will take out five of the contigs and BLAST them. Note that you need to be connected to the internet to do this.

First, sort the table according to the contig length by clicking the **Length of consensus sequence** column header. Now find the contigs that are just above 10,000 nucleotides. Select the five first as shown in figure 2.15.

Then click the **Open Consensus** button at the bottom of the table. That will create a sequence list with the five contig (consensus) sequences that you have selected.

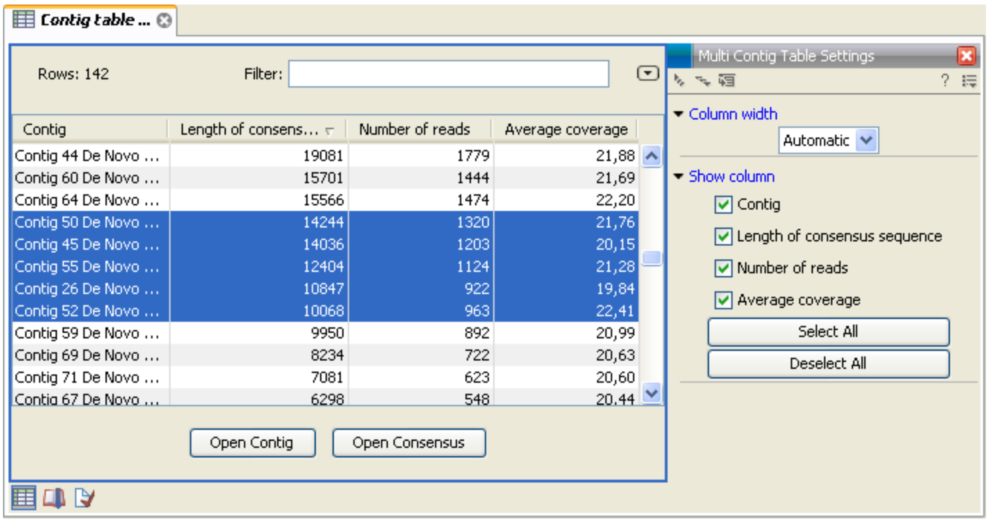
To BLAST these sequences, first make sure the sequence list with the five contigs has been **Saved** ( ). Then:

Toolbox | **BLAST Search ( )** | **NCBI BLAST ( )**

Select the sequence list that you have saved and click **Next**. To be completely open in terms of what kind of data there is in the sequences, choose the **Nucleotide collection** database (see figure 2.16).

Click **Next** and leave the settings at their default (to make sure the settings are back to default, click the **Reset to CLC Standard Settings** ( ) button). Click **Next** again and **Finish**

The result is shown in figure 2.17.



Contig table ...

Rows: 142 Filter:

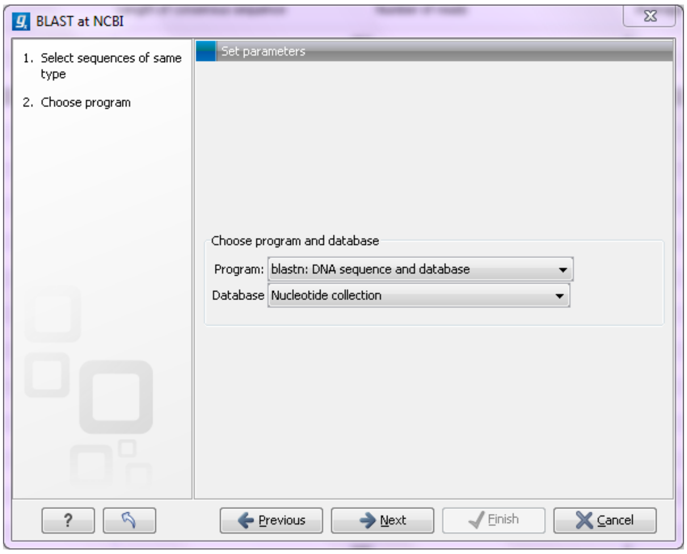
Contig	Length of consens...	Number of reads	Average coverage
Contig 44 De Novo ...	19081	1779	21,88
Contig 60 De Novo ...	15701	1444	21,69
Contig 64 De Novo ...	15566	1474	22,20
Contig 50 De Novo ...	14244	1320	21,76
Contig 45 De Novo ...	14036	1203	20,15
Contig 55 De Novo ...	12404	1124	21,28
Contig 26 De Novo ...	10847	922	19,84
Contig 52 De Novo ...	10068	963	22,41
Contig 59 De Novo ...	9950	892	20,99
Contig 69 De Novo ...	8234	722	20,63
Contig 71 De Novo ...	7081	623	20,60
Contig 67 De Novo ...	6298	548	20,44

Multi Contig Table Settings

Column width: Automatic

Show column
- ☑ Contig
- ☑ Length of consensus sequence
- ☑ Number of reads
- ☑ Average coverage

Figure 2.15: *Selecting five contigs that are just above 10,000 nucleotides long.*



BLAST at NCBI

1. Select sequences of same type
2. Choose program

Set parameters

Choose program and database

Program: blastn: DNA sequence and database

Database Nucleotide collection

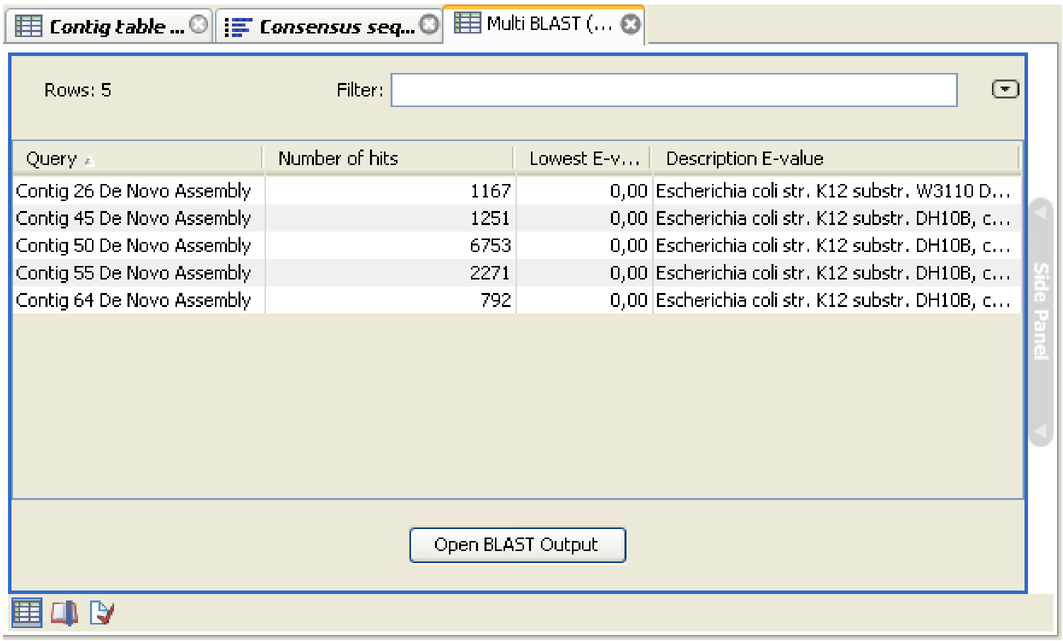
Figure 2.16: *Selecting the Nucleotide collection (nr) database.*

For each contig, you can see the description of the top hit which is *E. coli* str. K12 in all cases. If you wish to inspect the individual BLAST results in more detail, simply double-click the row and the BLAST result will open. If you wish to use the result for reporting, e.g. in a spread-sheet, you can **Copy** (📋) the contents of the table and paste into e.g. a spreadsheet for further processing. To select all the rows in the table, press Ctrl + A (⌘ +A on Mac).

You can also use the filter at the top to sort the table (e.g. to only show BLAST results with an E-value lower than a certain threshold).

2.6 Resequencing tutorial: Map reads to reference followed by SNP and DIP detection

This tutorial takes you through some of the tools for analyzing a typical resequencing data set from a high-throughput sequencing machine. As an example we use an *E. coli* data set consisting



Rows: 5 Filter:

Query	Number of hits	Lowest E-v...	Description E-value
Contig 26 De Novo Assembly	1167	0,00	Escherichia coli str. K12 substr. W3110 D...
Contig 45 De Novo Assembly	1251	0,00	Escherichia coli str. K12 substr. DH10B, c...
Contig 50 De Novo Assembly	6753	0,00	Escherichia coli str. K12 substr. DH10B, c...
Contig 55 De Novo Assembly	2271	0,00	Escherichia coli str. K12 substr. DH10B, c...
Contig 64 De Novo Assembly	792	0,00	Escherichia coli str. K12 substr. DH10B, c...

Figure 2.17: *The overview BLAST table.*

of a little more than 400,000 reads from a 454 sequencer. [3]

2.6.1 Importing the data

First, download the data set from our web site: `http://download.clcbio.com/testdata/raw_data/454.zip`. Unzip the file somewhere on your computer (e.g. the Desktop).

Start the *CLC Genomics Workbench* and import the data:

File | **Import High-throughput Sequencing** | **Roche 454**

This will bring up the dialog shown in figure 2.18

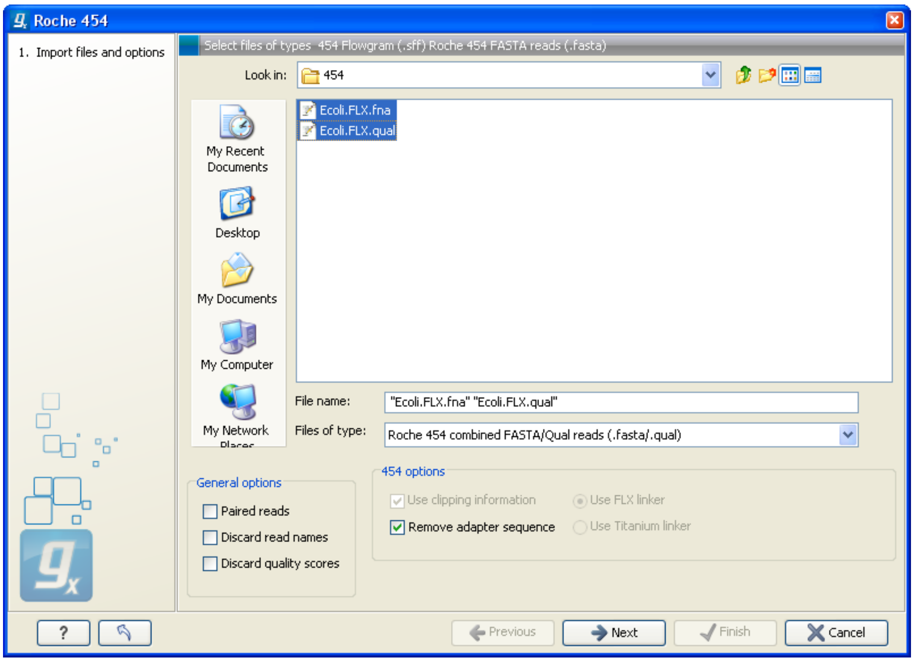
Select the `Ecoli.FLX.fna` and `Ecoli.FLX.qual` files that come from the downloaded zip file. Make sure the **Remove adapter sequence** checkbox is checked and that the **Paired reads** checkbox is NOT checked. The option to discard read names is not significant in this context because of the relatively small amount of reads. Click **Next**, choose to **Save** and click **Finish**.

After a short while, the reads have been imported. Next, import the reference genome sequence also included in the zip file:

File | **Import (📂)** | **Locate "NC_010473.gbk"** | **Select**

Note that this is a genbank file imported using the "normal" import tool. Next-generation sequencing data needs to be imported using the special tool in the toolbox because they have a more complex structure (e.g. in this case two files - one with sequence and one with quality scores).

[3] Note that there are special system requirements for *CLC Genomics Workbench*, see `http://www.clcbio.com/index.php?id=72`



Roche 454

1. Import files and options

Select files of types 454 Flowgram (.sff) Roche 454 FASTA reads (.fasta)

Look in: 454

Ecoli.FLX.fna
Ecoli.FLX.qual

File name: "Ecoli.FLX.fna" "Ecoli.FLX.qual"

Files of type: Roche 454 combined FASTA/Qual reads (.fasta/.qual)

General options

□ Paired reads
□ Discard read names
□ Discard quality scores

454 options

☑ Use clipping information
☑ Remove adapter sequence
○ Use FLX linker
○ Use Titanium linker

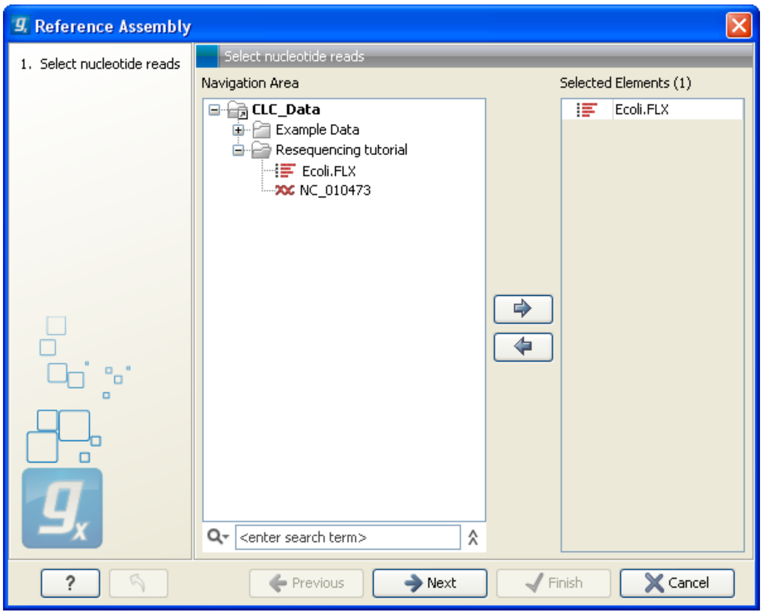
Figure 2.18: *Choosing what kind of data you wish to import.*

2.6.2 Mapping the reads to the reference

First step in the analysis is to map the reads to the reference genome:

> **Select the Ecoli.FLX sequence list | Toolbox | High-throughput Sequencing () | Map Reads to Reference ()**

This shows the dialog in figure 2.19).



Reference Assembly

1. Select nucleotide reads

Select nucleotide reads

Navigation Area

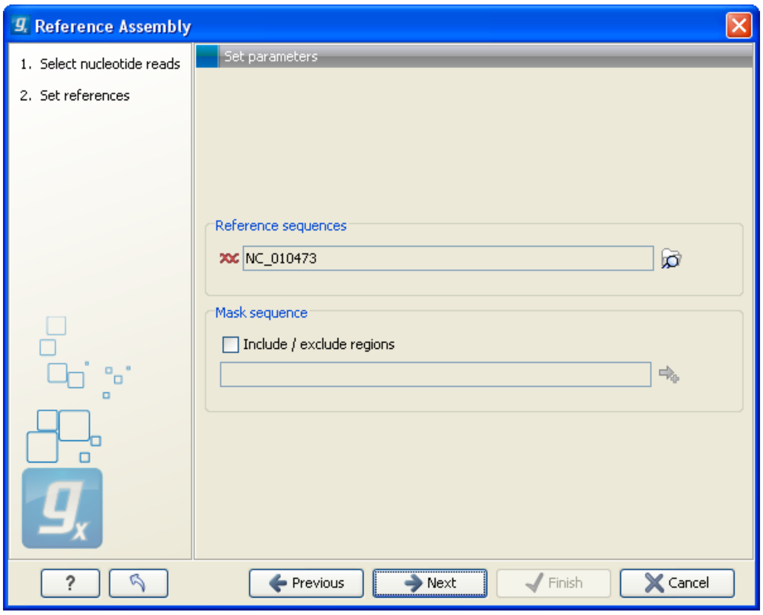
- CLC_Data
  - Example Data
  - Resequencing tutorial
    - Ecoli.FLX
    - NC_010473

Selected Elements (1)

Ecoli.FLX

Figure 2.19: *Select sequence list containing the reads. The reference sequence will be selected in the next step.*

Select the `Ecoli.FLX` () sequence list and add it to the panel to the right. Clicking **Next** will allow you to select a reference sequence as shown in figure 2.20.



Reference Assembly

1. Select nucleotide reads
2. Set references

Set parameters

Reference sequences

NC_010473

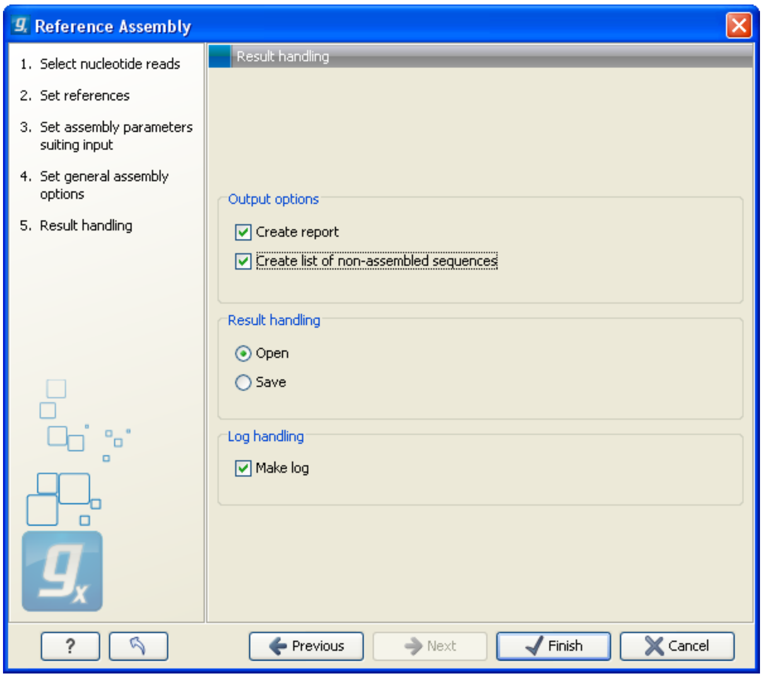
Mask sequence

☐ Include / exclude regions

Figure 2.20: *Specifying the reference sequences and masking.*

At the top you select the `NC_010473` () by clicking the **Browse and select element** () button. You can select either single sequences or a list of sequences as reference sequences, but in this case just select this single genome sequence.

Clicking **Next** until you see the dialog shown in figure 2.21 (for more information about the other settings, please click the **Help** (?) button).



Reference Assembly

1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input
4. Set general assembly options
5. Result handling

Result handling

Output options

☑ Create report

☑ Create list of non-assembled sequences

Result handling

● Open

○ Save

Log handling

☑ Make log

Figure 2.21: *Output options.*

Here you can select between different output options. Choose to **Create report** and **Create list**

of non-mapped reads. Click **Finish**.

2.6.3 Interpreting the read mapping results

You can follow the mapping progress both in the status bar at the bottom left corner and the log (if you have chosen to show the log in figure 2.21). When the process is done, you will see the following results:

List of non-mapped reads These are the reads that did not match the reference sequence. You can use this list to investigate contamination in the sample or structural differences between the sequencing data and the reference sequence. Typically you will do a *de novo* assembly of these reads and then use BLAST to investigate the contigs (there is a separate tutorial showing how to do this).

Report The report shows information about the mapping. Most importantly, it shows the number of reads that matched the reference sequence.

Mapping The mapping itself shows the alignment of all the reads to the reference. We are going into details with the mapping now.

The mapping (shown in figure 2.22) displays the reads including quality scores. For annotated reference sequences like this, you can display the translation of the coding regions (the yellow CDS annotations) in the Side Panel in the **Nucleotide info** group under **Translation**. (If you scroll a little to the right, you can see the first translation).

You can use the zoom buttons for zooming in and out, and you can scroll along the reference (both horizontally and vertically to inspect the reads covering the region of the view). The **Side panel** includes a lot of settings to change the appearance of the mapping, and there are a few that are particularly good to know:

Sequence layout – Compactness This is used to determine the height of the reads (show/hide the sequence and quality information). This is very useful to get an overview when the reads themselves are not important.

Alignment info – Coverage Displays a graph of the coverage. There are more relevant graphs under the **Alignment info** group.

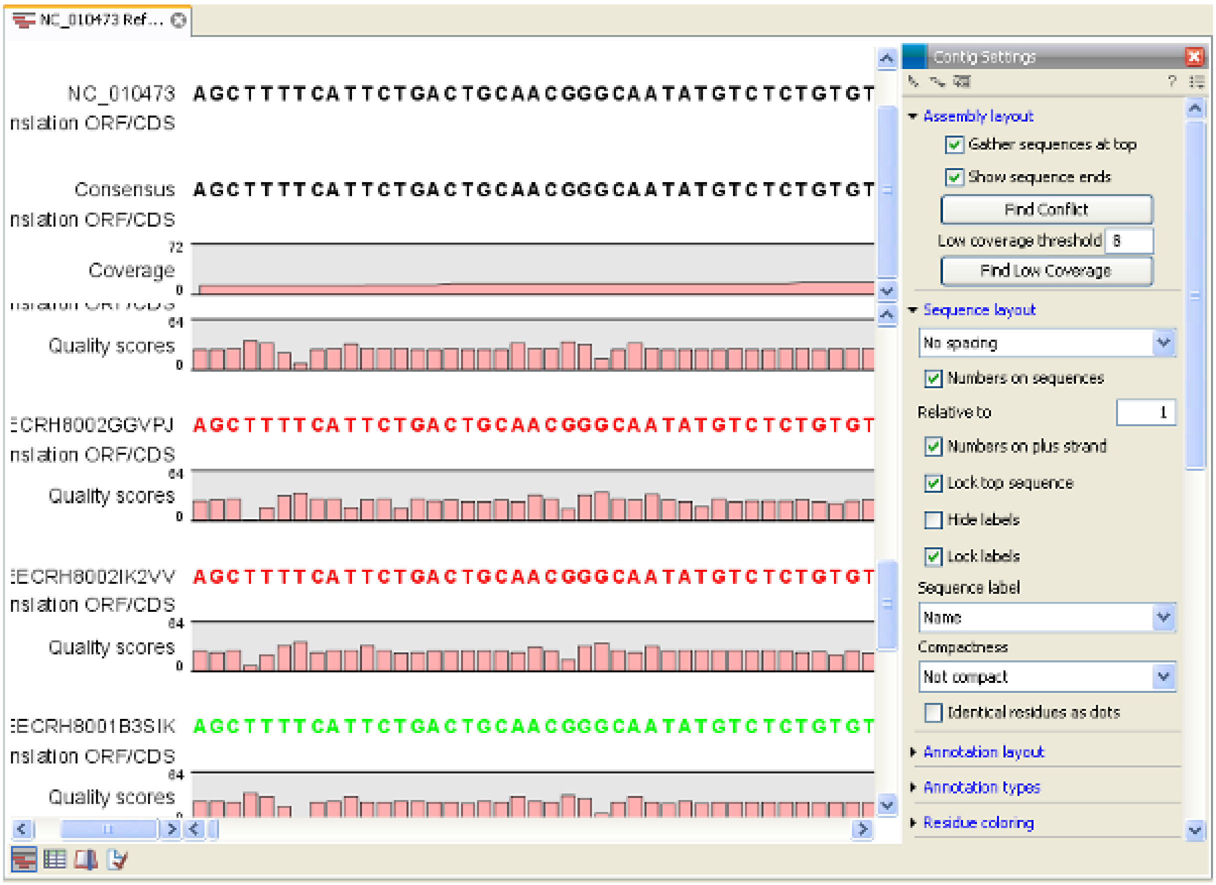
The read colors are green (forward) and red (reverse) by default.

Before we continue be sure to **Save** (🖫) the results.

2.6.4 Looking for SNPs

One way of inspecting the result is of course to go through the mapping manually, guided by the annotations on the reference sequence. Given the huge amounts of data, manual inspection is only advisable if you have special loci that you are interested in. For a more systematic approach, the *CLC Genomics Workbench* provides two tools: SNP and DIP detection to help you get an overview of the differences between the reference sequence and the reads.

We first look at SNP detection:


NC_010473 Ref...
NC_010473 AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGT
nslation ORF/CDS
Consensus AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGT
nslation ORF/CDS
72
Coverage
0
64
Quality scores
0
ECRH8002GGVPJ AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGT
nslation ORF/CDS
64
Quality scores
0
EECRH8002IK2VV AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGT
nslation ORF/CDS
64
Quality scores
0
EECRH8001B3SIK AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGT
nslation ORF/CDS
64
Quality scores
0
Contig Settings

Assembly layout
- ☑ Gather sequences at top
- ☑ Show sequence ends
- Find Conflict
- Low coverage threshold 8
- Find Low Coverage

Sequence layout
- No spacing
- ☑ Numbers on sequences
- Relative to 1
- ☑ Numbers on plus strand
- ☑ Lock top sequence
- ☐ Hide labels
- ☑ Lock labels
- Sequence label: Name
- Compactness: Not compact
- ☐ Identical residues as dots

Annotation layout

Annotation types

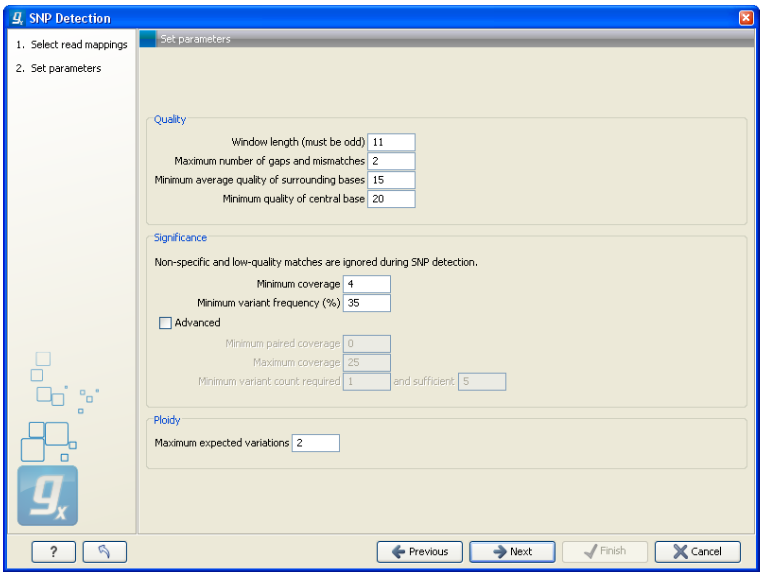
Residue coloring

Figure 2.22: *The reads mapped to the reference.*

Toolbox | **High-throughput Sequencing** () | **SNP detection** ()

This opens a dialog where you can select the mapping result () you saved before.

Clicking **Next** will display the dialog shown in figure 2.23



Quality
- Window length (must be odd) 11
- Maximum number of gaps and mismatches 2
- Minimum average quality of surrounding bases 15
- Minimum quality of central base 20

Significance

Non-specific and low-quality matches are ignored during SNP detection.
- Minimum coverage 4
- Minimum variant frequency (%) 35
- ☐ Advanced
- Minimum paired coverage 0
- Maximum coverage 25
- Minimum variant count required 1 and sufficient 5

Ploidy
- Maximum expected variations 2

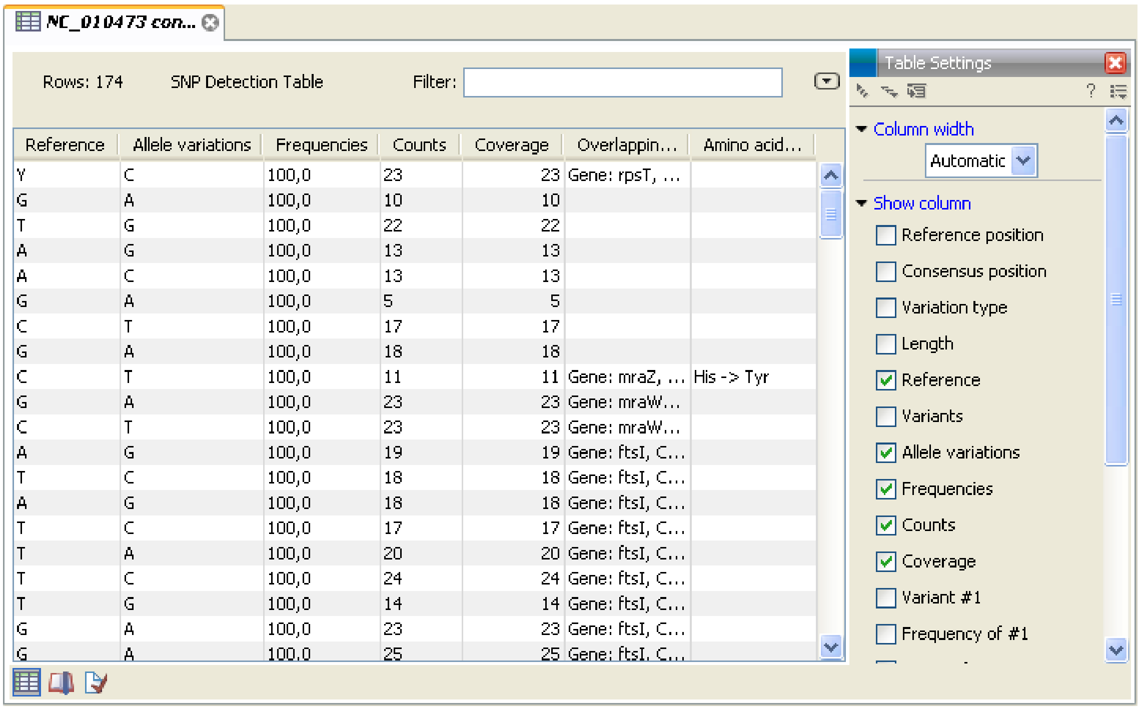
Figure 2.23: *SNP detection parameters.*

The default settings work well for this data set, but it is important to notice two of the settings that you should always consider when working with your own data: If the **Minimum coverage** is set to 4 but you have a mapping with an average coverage of 2, a lot of potential SNPs will not be reported. The **Minimum variant frequency** should also be set, especially when working with mixed samples or non-haploid organisms (e.g. for diploids, it should be set below 50 % in order to report heterozygote SNPs). For this analysis we leave the setting at 35 %.

For more information about the settings, click the **Help** (?) button.

Click **Next** and **Finish**.

The result of the SNP detection will now open a table as displayed in figure 2.24 below the mapping.



Rows: 174 SNP Detection Table Filter:

Reference	Allele variations	Frequencies	Counts	Coverage	Overlappin...	Amino acid...
Y	C	100,0	23	23	Gene: rpsT, ...	
G	A	100,0	10	10		
T	G	100,0	22	22		
A	G	100,0	13	13		
A	C	100,0	13	13		
G	A	100,0	5	5		
C	T	100,0	17	17		
G	A	100,0	18	18		
C	T	100,0	11	11	Gene: mraZ, ...	His -> Tyr
G	A	100,0	23	23	Gene: mraW...	
C	T	100,0	23	23	Gene: mraW...	
A	G	100,0	19	19	Gene: ftsI, C...	
T	C	100,0	18	18	Gene: ftsI, C...	
A	G	100,0	18	18	Gene: ftsI, C...	
T	C	100,0	17	17	Gene: ftsI, C...	
T	A	100,0	20	20	Gene: ftsI, C...	
T	C	100,0	24	24	Gene: ftsI, C...	
T	G	100,0	14	14	Gene: ftsI, C...	
G	A	100,0	23	23	Gene: ftsI, C...	
G	A	100,0	25	25	Gene: ftsI, C...	

Table Settings

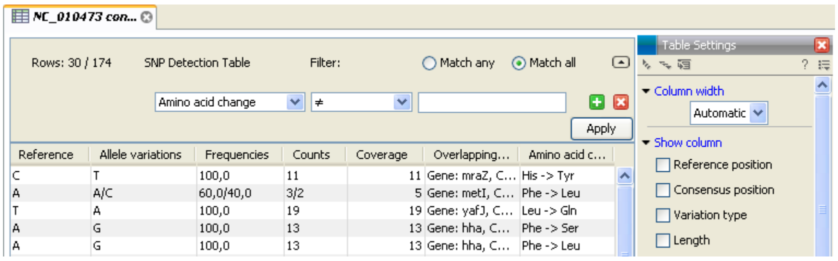
Column width: Automatic

Show column
- □ Reference position
- □ Consensus position
- □ Variation type
- □ Length
- ☑ Reference
- □ Variants
- ☑ Allele variations
- ☑ Frequencies
- ☑ Counts
- ☑ Coverage
- □ Variant #1
- □ Frequency of #1

Figure 2.24: *A table of SNPs.*

By clicking in the table, you can browse through the SNPs. You can **Copy** ( ) the contents of the table and paste into e.g. a spreadsheet for further processing. To select all the rows in the table, press Ctrl + A (⌘ +A on Mac). You can also **Export** ( ) the table in Excel format.

You can also use the filter at the top to sort the table, e.g. to show only nonsynonymous SNPs (filter the **Amino acid change** column to not being empty as shown in figure 2.25).



Rows: 30 / 174 SNP Detection Table Filter: ○ Match any ● Match all

Amino acid change ≠

Reference	Allele variations	Frequencies	Counts	Coverage	Overlapping...	Amino acid c...
C	T	100,0	11	11	Gene: mraZ, C...	His -> Tyr
A	A/C	60,0/40,0	3/2	5	Gene: metI, C...	Phe -> Leu
T	A	100,0	19	19	Gene: yafJ, C...	Leu -> Gln
A	G	100,0	13	13	Gene: hha, C...	Phe -> Ser
A	G	100,0	13	13	Gene: hha, C...	Phe -> Leu

Table Settings

Column width: Automatic

Show column
- □ Reference position
- □ Consensus position
- □ Variation type
- □ Length

Figure 2.25: *Filtering the SNP table to only display nonsynonymous SNPs.*

2.6.5 Looking for DIPs

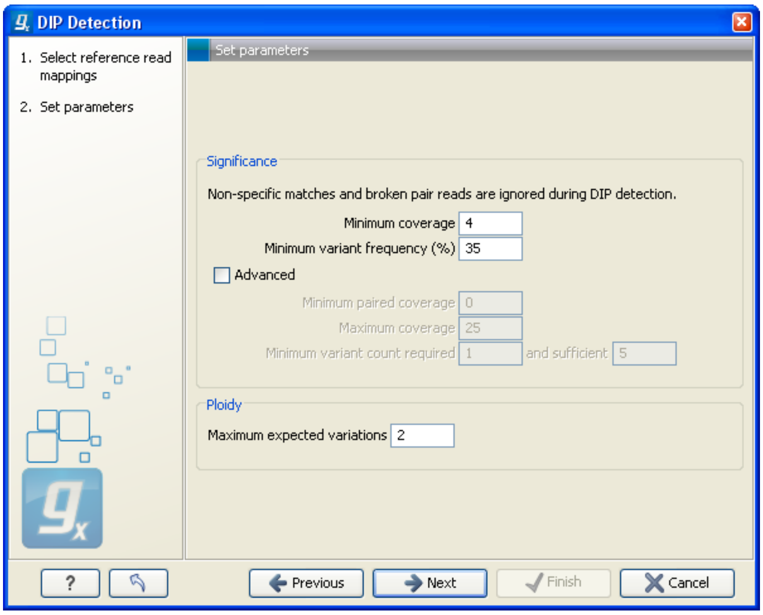
Besides looking for SNPs, you can also do a systematic search for DIPs (Deletion-Insertion Polymorphisms). If you have high coverage in your mapping, you will often find a lot of gaps in the consensus sequence. This is because just a single insertion in one of the reads will cause a gap in all other sequences at this position. The majority of all these gaps should simply be ignored, because they are introduce by single or very few reads because of sequencing errors. The automated DIP detection can be used to find the ones that are significant.

If you want to use the consensus sequence for other purposes, you can simply ignore all the gaps (they will disappear once the consensus sequence gets out of the mapping), and the significant ones can then be annotated as DIPs:

Toolbox | **High-throughput Sequencing ( )** | **DIP detection ( )**

This opens a dialog where you can select the mapping result ( ) you saved before.

Clicking **Next** will display the dialog shown in figure 2.26



Non-specific matches and broken pair reads are ignored during DIP detection.

Minimum coverage 4

Minimum variant frequency (%) 35

□ Advanced

Minimum paired coverage 0

Maximum coverage 25

Minimum variant count required 1 and sufficient 5

Maximum expected variations 2

Figure 2.26: *DIP detection parameters.*

The default settings work well for this data set.

For more information about the settings, click the **Help** (?) button.

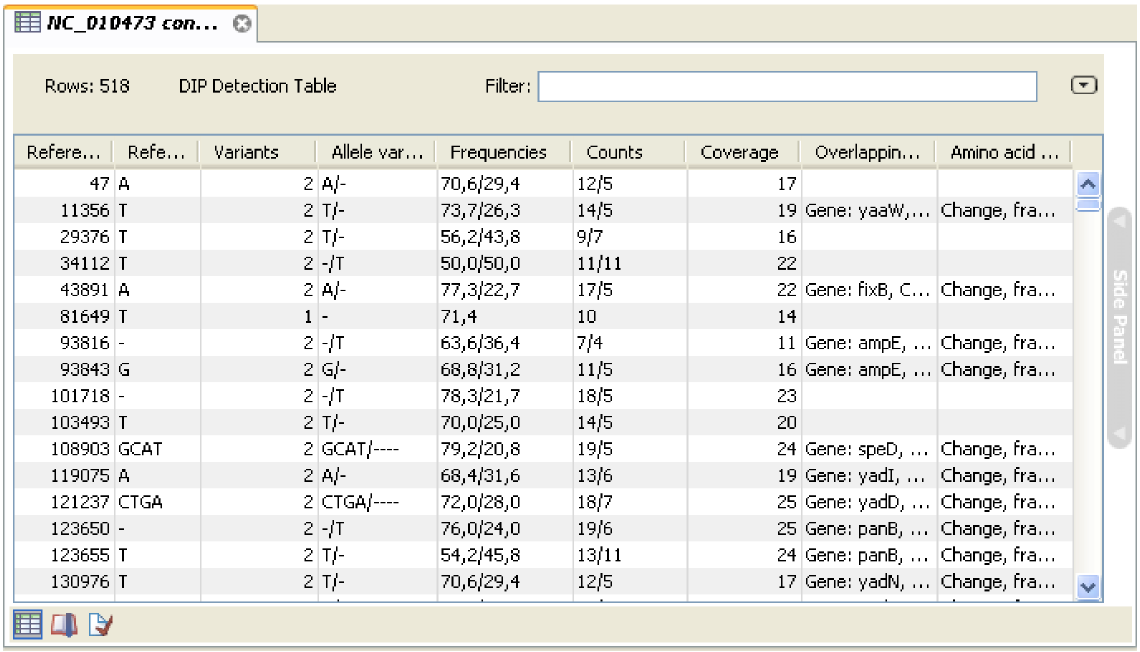
Click **Next** and **Finish**.

The result of the DIP detection will now open a table as displayed in figure 2.27 below the mapping.

By clicking in the table, you can browse through the DIPs. You can **Copy** ( ) the contents of the table and paste into e.g. a spreadsheet for further processing. To select all the rows in the table, press Ctrl + A (⌘ +A on Mac).

You can also use the filter at the top to sort the table.

Note: 454 data has an inherent flaw when it comes to homopolymer stretches. This means that



Rows: 518 DIP Detection Table Filter:

Refere...	Refe...	Variants	Allele var...	Frequencies	Counts	Coverage	Overlappin...	Amino acid ...
47	A	2	A/-	70,6/29,4	12/5	17		
11356	T	2	T/-	73,7/26,3	14/5	19	Gene: yaaW,...	Change, fra...
29376	T	2	T/-	56,2/43,8	9/7	16		
34112	T	2	-/T	50,0/50,0	11/11	22		
43891	A	2	A/-	77,3/22,7	17/5	22	Gene: fixB, C...	Change, fra...
81649	T	1	-	71,4	10	14		
93816	-	2	-/T	63,6/36,4	7/4	11	Gene: ampE, ...	Change, fra...
93843	G	2	G/-	68,8/31,2	11/5	16	Gene: ampE, ...	Change, fra...
101718	-	2	-/T	78,3/21,7	18/5	23		
103493	T	2	T/-	70,0/25,0	14/5	20		
108903	GCAT	2	GCAT/----	79,2/20,8	19/5	24	Gene: speD, ...	Change, fra...
119075	A	2	A/-	68,4/31,6	13/6	19	Gene: yadI, ...	Change, fra...
121237	CTGA	2	CTGA/----	72,0/28,0	18/7	25	Gene: yadD, ...	Change, fra...
123650	-	2	-/T	76,0/24,0	19/6	25	Gene: panB, ...	Change, fra...
123655	T	2	T/-	54,2/45,8	13/11	24	Gene: panB, ...	Change, fra...
130976	T	2	T/-	70,6/29,4	12/5	17	Gene: yadN, ...	Change, fra...

Figure 2.27: *A table of DIPs.*

the exact number of bases reported in a homopolymer stretch is not always right. This gives rise to a lot of small insertions and deletions in a data set like this. When you browse through the DIPs in the table, you will see that most of them are single-base insertions or deletions at the end of a homopolymer stretch. You can sort by length to see some of the longer and more interesting insertions/deletions.

2.7 Map reads to reference in details

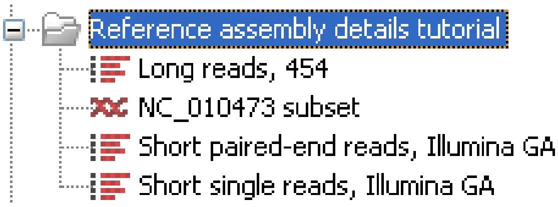
This tutorial goes into detail with read mapping of the *CLC Genomics Workbench*. Whereas the Resequencing tutorial shows the overall work flow including read mapping, this tutorial will go into much more detail about the mapping algorithm. You will learn what the parameters mean, and you will hopefully end up being able to make better decisions about the analysis of your own data.

We recommend going through the resequencing tutorial first, since it includes the basic introduction to this topic. Find it at `http://www.clcbio.com/tutorials`.

In this tutorial we use a subset of an *E. coli* data set where we focus on a small part of the genome and use a combination of 454 and Illumina GA reads.

2.7.1 Importing the data

First, download the data set from our web site: `http://download.clcbio.com/testdata/Read_mapping_details_tutorial.zip`. Start the *CLC Genomics Workbench* and **Import** ( ) the file. You will now have a folder structure like the one shown in figure 2.28.



Reference assembly details tutorial
- Long reads, 454
- NC_010473 subset
- Short paired-end reads, Illumina GA
- Short single reads, Illumina GA

Figure 2.28: *The data set consisting of reference sequence (subset of genome) and three sets of sequencing reads.*

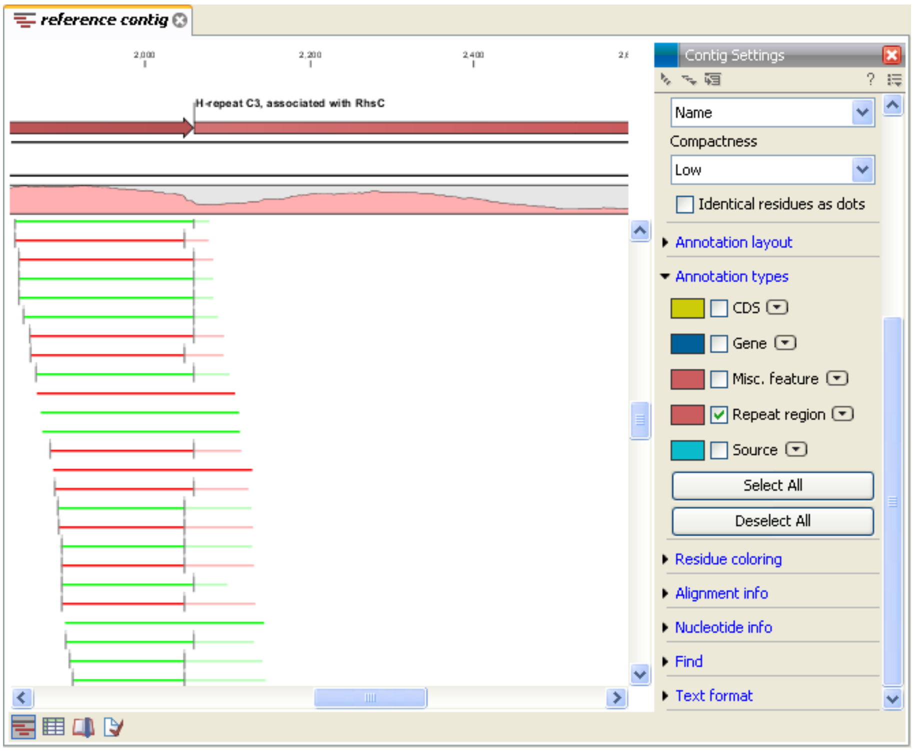
2.7.2 Mapping long reads

The *CLC Genomics Workbench* uses two different algorithms for mapping reads to a reference depending on the read length. First, we consider long reads. Reads are defined as "long" when they are longer than 55 bp.

First, map the reads with standard settings using the `Long reads, 454` as input and the `NC_010473 subset` as reference. You can set the parameters to default by clicking the button ( ) at the bottom of the dialog. You do not need to create a log in the last step before you click **Finish**.

Change the compactness in the **Side Panel** to **Low** and deselect all **Annotation types** except **Repeat region**. To save these adjustments so that they take effect next time you open a mapping, click the **Save/Restore Settings** button ( ) at the top of the **Side Panel** and click **Save Settings**. Give your settings a name and make sure the check box to **Always apply these settings** is checked.

Now zoom out so that you have a view similar to the one shown in figure 2.29 and go to around position 2060.



Compactness: Low

□ Identical residues as dots

Annotation types:
- 🟨 □ CDS
- 🟦 □ Gene
- 🟥 □ Misc. feature
- 🟥 ☑ Repeat region
- 🟦 □ Source

Figure 2.29: *Mapping reads to a repeat region.*

Note that there is something special going on just at the border between the two annotated repeat regions. If you **Zoom in** () to see the details, you can see that for many reads, the end of the read is faded. This is because this part of the read was not aligned to the reference during mapping. On closer inspection you can see that this part of the read doesn't match the reference sequence (figure 2.30).

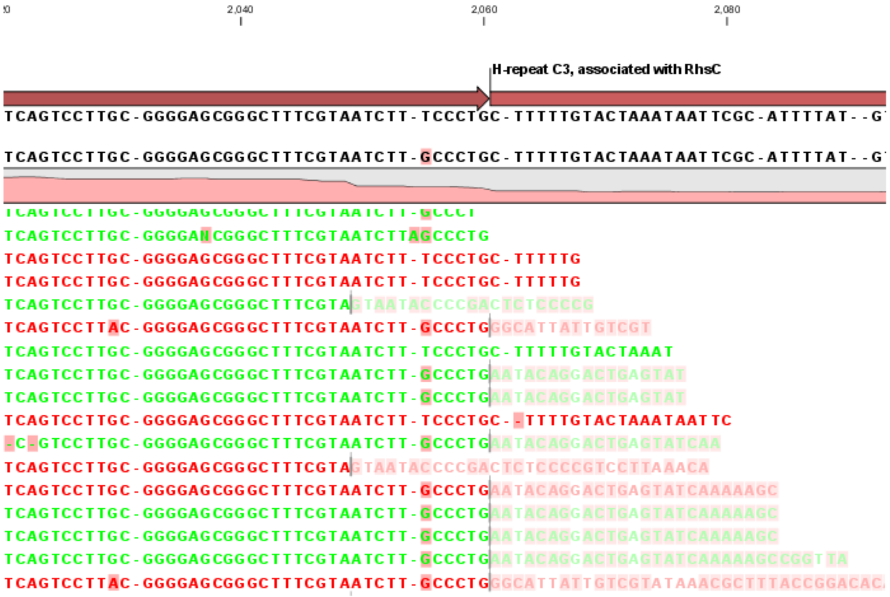

2,040
2,060
2,080
H-repeat C3, associated with RhsC
TCAGTCCTTGC-GGGGAGCGGGCTTTCGTAATCTT-TCCCTGC-TTTTTGTACTAAATAATTCGC-ATTTTAT--G
TCAGTCCTTGC-GGGGAGCGGGCTTTCGTAATCTT-GCCCTGC-TTTTTGTACTAAATAATTCGC-ATTTTAT--G

Figure 2.30: *Unaligned ends.*

The reason why you see this is because the *CLC Genomics Workbench* performs *local alignment* of the reads. This means that it only aligns the part of the read that matches well – the rest is left unaligned.

Mapping long reads in *CLC Genomics Workbench* is a two-step process:

1. For each read, the optimal local alignment between the read and the reference sequence is found.

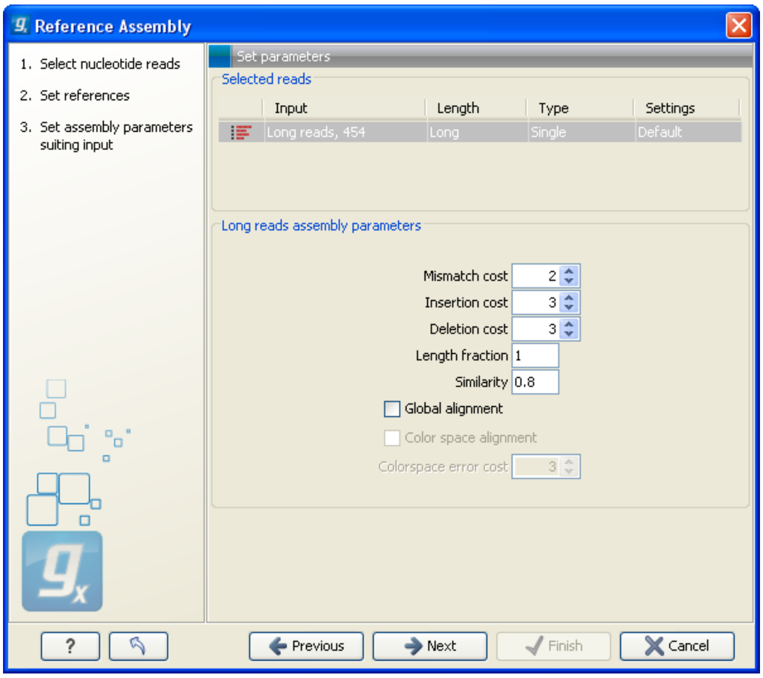
2. Second, all reads are filtered according to user-defined criteria for length and similarity of the local alignment

To illustrate this, repeat the read mapping but this time adjust the **Length fraction** to 1 as shown in figure 2.31.

This means that the reads will be filtered so that only reads matching in their entire length will be included in the mapping. Finish the read mapping with this setting and see how it affects the same region. As expected, all the reads with un-aligned ends are gone (see figure 2.32), and you are left with less reads (the coverage is around half compared to figure 2.29).

If you go back and look at figure 2.30, you can see that besides the unaligned ends, there are also a number of mismatches in the reads. These mismatches are internal to the local alignment, whereas the unaligned ends were outside the local alignment.

Just as with the unaligned ends, you can control the level of similarity between reads and reference using the **Similarity** setting (see figure 2.31). Run the mapping again, this time with



Reference Assembly

1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input

Set parameters

Selected reads

	Input	Length	Type	Settings
	Long reads, 454	Long	Single	Default

Long reads assembly parameters

Mismatch cost 2

Insertion cost 3

Deletion cost 3

Length fraction 1

Similarity 0.8

□ Global alignment

□ Color space alignment

Colorspace error cost 3

Figure 2.31: *Stricter alignment settings.*

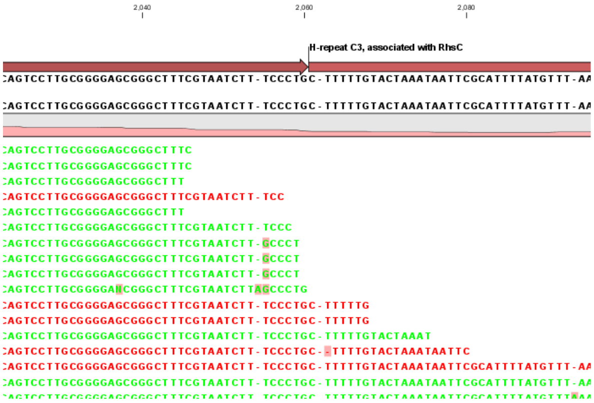
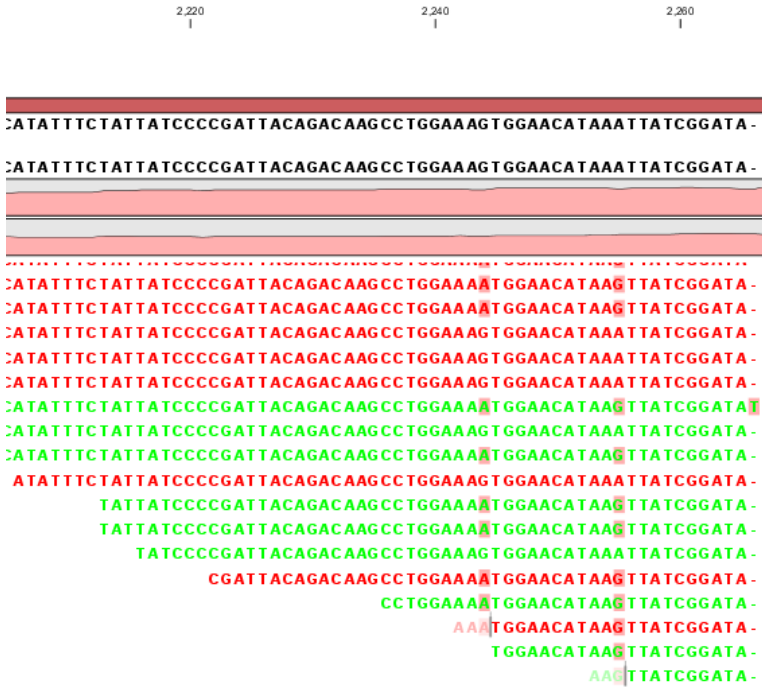


Figure 2.32: *Stricter alignment settings result in fewer reads mapped.*

a **Similarity** of 1. Set the **Length fraction** back to 0.5. You would now expect to have removed all the reads that contain mismatches. But if you look at the result, you will see that this is not the case. Figure 2.33 shows an example around position 2250, where several reads contain mismatches.

The reason why you see this is that we have set the **Length fraction** back to 0.5. This means that we have accepted that *half the read should match perfectly to the reference*. The other half is allowed to diverge from the reference.

Besides the **Length fraction** and **Similarity**, you can also specify gap and mismatch costs (see figure 2.31). These determine how the initial local alignment should be performed. Setting a low mismatch cost and high insertion/deletion costs will favor mismatches over gaps in the local alignment. We are not going into details with these settings in this tutorial.



```
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-

CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-

CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATAT
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
CATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
 ATATTTCTATTATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
        TATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
        TATTATCCCCGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
           TATCCCCGATTACAGACAAGCCTGGAAAGTGGAACATAAATTATCGGATA-
                 CGATTACAGACAAGCCTGGAAAATGGAACATAAGTTATCGGATA-
                               CCTGGAAAATGGAACATAAGTTATCGGATA-
                                     AAATGGAACATAAGTTATCGGATA-
                                        TGGAACATAAGTTATCGGATA-
                                                AAGTTATCGGATA-
```

Figure 2.33: *Still mismatches although similarity was set to 1.*

2.7.3 Mapping short reads

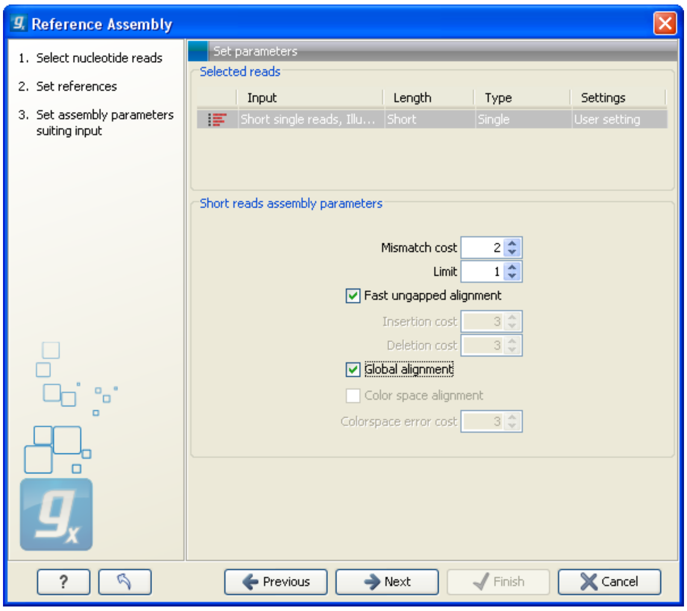
As mentioned in the beginning, the *CLC Genomics Workbench* uses two different algorithms for mapping, depending on the read length. We will now take a look at the short reads. In order to show how different settings affect the result, we are going to perform one round of read mapping with very strict settings, requiring a perfect match, and we will then take the unmapped reads and map them with more relaxed settings. First, close the open views by pressing Ctrl + Shift + W (⌘ + Shift + W on Mac). You do not need to save the results.

Select the `Short single reads, Illumina GA` and open the read mapping dialog. Use the same reference sequence as before. In the third step, set **Limit** to 1 and check the **Global alignment** checkbox (see figure 2.34). Note that we will come back to have a closer look at these settings later, but to summarize it means that we require a perfect match for the full read.

Go to the last step and check the **Create list of non-mapped reads**. The result of the mapping will this time be the mapping itself and a list of the reads that didn't match the reference. Save this list and do a new round of read mapping with this list as input. This time, raise the **Limit** to the maximum of 11, uncheck **Fast ungapped alignment** (see figure 2.35) and uncheck **Global alignment**. This is the most relaxed setting we can make which allows mismatches, gaps and unaligned ends (explained in detail below).

We now have a mapping with reads that do not match the reference perfectly. If you go to the same position on the reference sequence as we examined before (2060), you can see what this means for the alignment of the reads (see figure 2.36).

First of all, this means that some reads have mismatches (see the top read in figure 2.36 which has two mismatches). Second, it means that there are un-aligned ends just like we saw when aligning the long reads. With the strict mapping settings in figure 2.34, we selected the global alignment option which means that no unaligned ends are accepted, no matter how the Limit is set. But with the relaxed settings, we are doing local alignment (as you can see from figure 2.35).


Reference Assembly

1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input

Set parameters

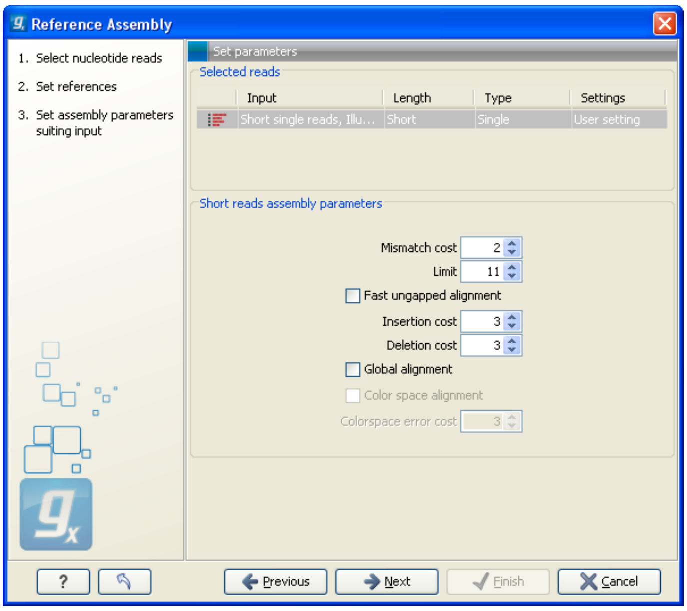
Selected reads

Input	Length	Type	Settings
Short single reads, Illu...	Short	Single	User setting

Short reads assembly parameters

Mismatch cost 2
Limit 1
☑ Fast ungapped alignment
Insertion cost 3
Deletion cost 3
☑ Global alignment
□ Color space alignment
Colorspace error cost 3

Figure 2.34: *Very strict alignment settings for short reads.*


Reference Assembly

1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input

Set parameters

Selected reads

Input	Length	Type	Settings
Short single reads, Illu...	Short	Single	User setting

Short reads assembly parameters

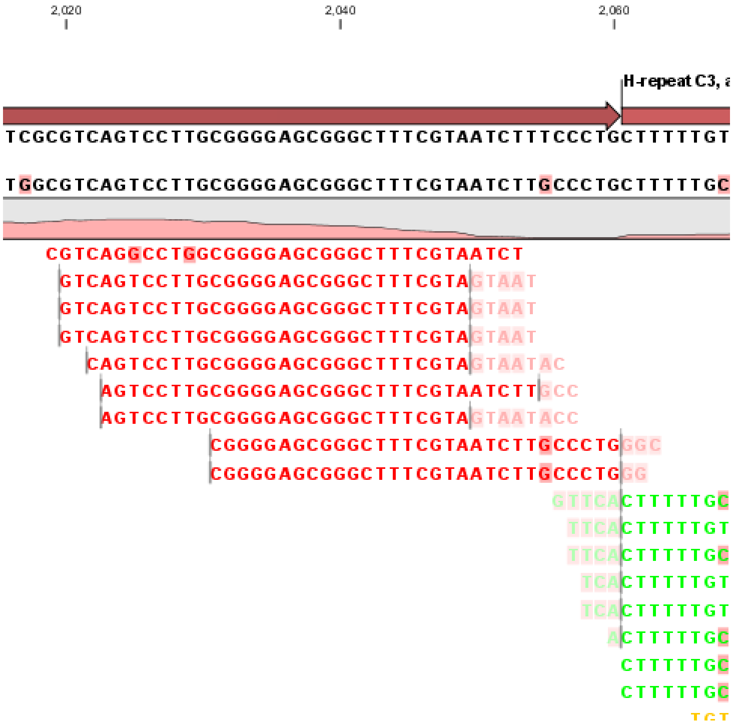
Mismatch cost 2
Limit 11
□ Fast ungapped alignment
Insertion cost 3
Deletion cost 3
□ Global alignment
□ Color space alignment
Colorspace error cost 3

Figure 2.35: *Very relaxed alignment settings for short reads.*

To explain what kind of mismatches and unaligned ends that are accepted, we need to inspect the details of the short read mapping algorithm. It is based on a scoring system where a match is rewarded one point. The other scores for mismatch, insertion and deletion can be adjusted in the dialog (see figure 2.35). The defaults are:

Match	$+1$ (cannot be changed)
Mismatch	-2
Deletion	-3
Insertion	-3

Let's take a few examples and calculate the score of an alignment. Below is a read of 20 bp perfectly aligned to the reference. The score is thus 20.



```
TCGCGTCAGTCCTTGCGGGGAGCGGGCTTTCGTAATCTTTCCCTGCTTTTTGT

TGGCGTCAGTCCTTGCGGGGAGCGGGCTTTCGTAATCTTGCCCTGCTTTTTGC
   CGTCAGGCCTGGCGGGGAGCGGGCTTTCGTAATCT
    GTCAGTCCTTGCGGGGAGCGGGCTTTCGTAGTAAT
    GTCAGTCCTTGCGGGGAGCGGGCTTTCGTAGTAAT
    GTCAGTCCTTGCGGGGAGCGGGCTTTCGTAGTAAT
      CAGTCCTTGCGGGGAGCGGGCTTTCGTAGTAATAC
       AGTCCTTGCGGGGAGCGGGCTTTCGTAATCTTGCC
       AGTCCTTGCGGGGAGCGGGCTTTCGTAGTAATACC
               CGGGGAGCGGGCTTTCGTAATCTTGCCCTGGGC
               CGGGGAGCGGGCTTTCGTAATCTTGCCCTGGG
                                        GTTCACTTTTTGC
                                         TTCACTTTTTGT
                                         TTCACTTTTTGC
                                          TCACTTTTTGT
                                          TCACTTTTTGT
                                            ACTTTTTGC
                                             CTTTTTGC
                                             CTTTTTGC
                                                  TGT
```

Figure 2.36: *Alignment of non-perfect matches.*

```
CGTATCAATCGATTACGCTATGAATG
   ||||||||||||||||||||        20
   ATCAATCGATTACGCTATGA
```

If we introduce a mismatch in the middle, we subtract 2 from the score. In addition, we also lose a match (we only have 19 matches left), so the score ends up at 17:

```
CGTATCAATCGATTACGCTATGAATG
   |||||||| |||||||||||        17
   ATCAATCGGTTACGCTATGA
```

If we have a mismatch near the end, we will simply exclude this base from the alignment. This means that we do not consider it as a mismatch but only one base less in the alignment. So the score ends at 19:

```
CGTATCAATCGATTACGCTATGAATG
    |||||||||||||||||||        19
   TTCAATCGATTACGCTATGA
```

As an exercise, we have listed a few alignments where you can calculate the score yourself (see answers in the foot note).[4]

[4] Answers: A=15, B=13, C=12

```
CGTATCAATCGATTACGCTATGAATG
    ||||||| ||||||||||         A
   CTCAATCGGTTACGCTATGA

CGTATCAATCGATTACGCTATGAATG
    ||||||| |||| ||||||        B
   TTCAATCGGTTACCCTATGA

CGTATCAATCGATTACGCTATGAATG
    ||||||| |||| |||||         C
   TTCAATCGGTTACCCTATGC
```

The alignment score is then compared to the read length, and the read is accepted if the difference does not exceed the **Limit** you specify in the dialog (e.g. 11 in figure 2.35). Example A above with a read length of 20 and a score of 15 would be accepted if the limit is 5 or higher. With the default settings where the limit is 8, it means that up to 2 mismatches are allowed or that 8 bases can be left un-aligned.

2.7.4 Making use of paired information

Perform a new round of read mapping with the `Short single reads, Illumina GA` data set using default parameters. You can set the parameters to default by clicking the button (↶) at the bottom of the dialog.

In the result scroll to around position 2100. Here you will see reads colored in yellow (see figure 2.37).

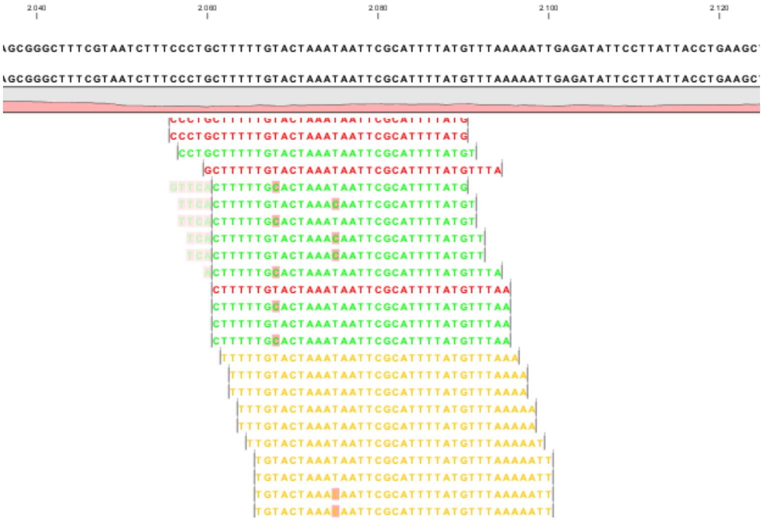
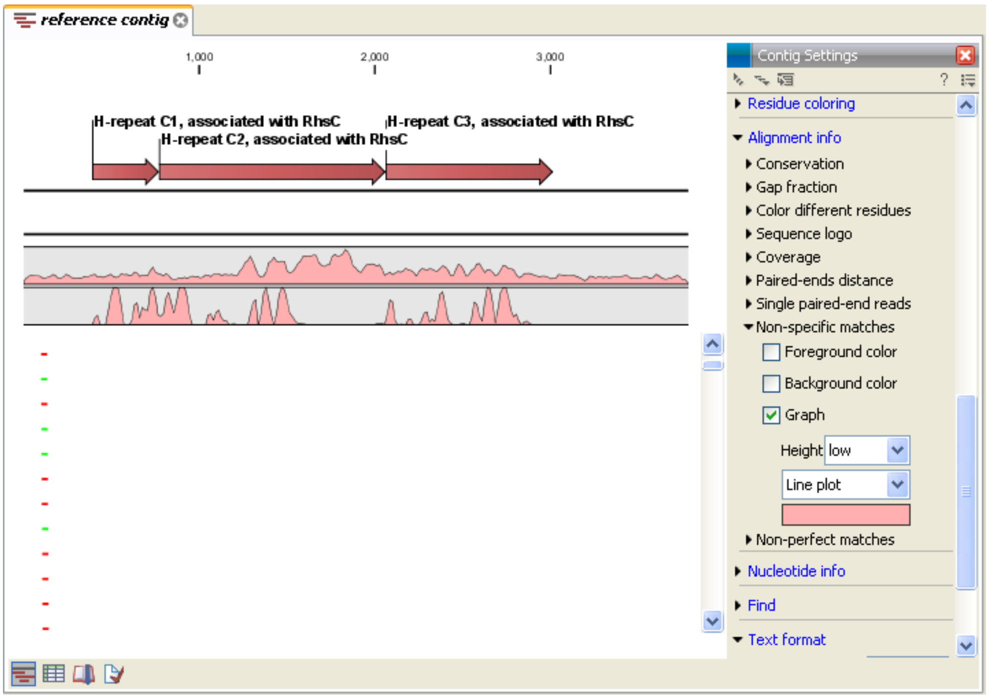


Figure 2.37: *Alignment of non-specific hits.*

The yellow color means that these reads could have been matched equally well other places on the reference sequence. They are what we call **Non-specific matches**. We see this because the reference sequence includes repetitive regions. You can show a graph of the frequency of non-specific matches in the **Side Panel** under **Alignment info**. Zoom out and you will see a view as shown in figure 2.38.



Non-specific matches

□ Foreground color

□ Background color

☑ Graph

Height low

Line plot

Figure 2.38: *Showing a graph of the frequency of non-specific hits.*

This graph shows the percentage of the aligned reads that are non-specific. You can **Zoom in** (🔍) by dragging a rectangle around one of the peaks to see the placement of the reads.

The reads in this data set are very short – only 35 bp. Until now we have been working with the `Short single reads, Illumina GA` data set, but let's see what happens if we do the same thing with the paired data set. Perform a new round of mapping with default parameters with the `Short paired-end reads, Illumina GA`. This data set is identical to the single reads data set except for the paired information. Display the **Non-specific reads** graph and zoom out. Your view should now look like figure 2.39.

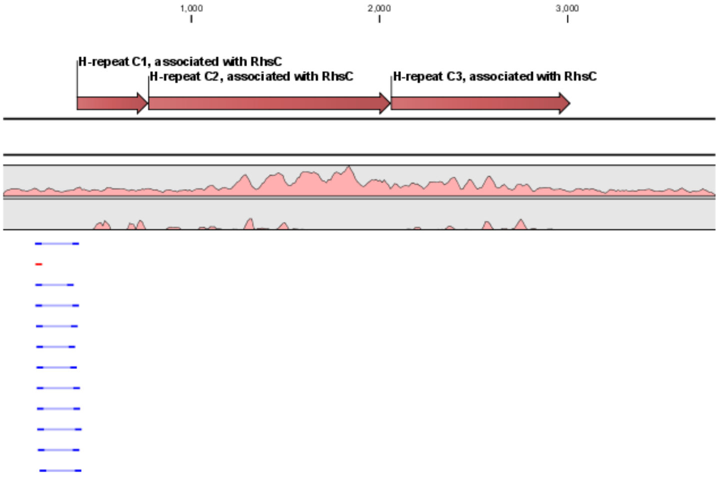

1,000
2,000
3,000
H-repeat C1, associated with RhsC
H-repeat C2, associated with RhsC
H-repeat C3, associated with RhsC

Figure 2.39: *The number of non-specific reads decreases dramatically when paired information is used.*

Compared with the single reads data set, there are considerably less non-specific reads in this paired data set. This is because the Workbench now has much more information that can be used to place the reads unambiguously: The paired read has 70 bp and spans a region of approximately 215 bp of the reference sequence. Although the total number of base pairs are the same in the two data sets, the extra information in the paired read structure produces a mapping of much higher quality.

The paired data can also be used to visualize genomic re-arrangements. You can read more about this in the user manual by searching for "Interpreting genomic re-arrangements".

2.8 ChIP sequencing: the basics

This tutorial takes you through a ChIP sequencing work flow using the *CLC Genomics Workbench*.

ChIP-Sequencing is used to analyze the interactions of proteins with genomic DNA. After a cross-binding step that tightly links proteins and DNA, ChIP-Seq uses chromatin immunoprecipitation (ChIP) to fish out the relevant pieces of genomic DNA. By subsequent massively parallel DNA sequencing and mapping to the reference genome it is possible to precisely identify binding sites of DNA-associated proteins. It can be used to precisely map global binding sites for any protein of interest but a practical limitation is the existence of good antibodies for the ChIP step. A natural next step bioinformatic analysis is to extract the binding regions and perform pattern discovery to learn about any conserved binding motif in the DNA. For further information, see the Wikipedia entry at http://en.wikipedia.org/wiki/Chip-Seq

For this tutorial, we use an artificial data set which was used for a ChIP-seq analysis competition announced at http://seqanswers.com/forums/showthread.php?t=1039, known as the "ChIP-seq challenge". We only use a subset of the original data set for this tutorial, since the purpose is to learn the basic principles.

The work flow consists of three parts: first, you import the data. Next, you map the reads to a reference. Finally, you use the ChIP sequencing tool to detect significant peaks in the sample.

In this tutorial we will not go through the details of the ChIP-seq analysis. The user manual already explains the details of the algorithm: Click the **Help** (?) button in the dialog (see below) to read this or go to http://www.clcbio.com/index.php?id=1330&manual=ChIP_sequencing.html.

2.8.1 Importing the data

First, download the data set from our web site: http://download.clcbio.com/testdata/raw_data/chip-seq.zip. Unzip the file somewhere on your computer (e.g. the Desktop).

Start the *CLC Genomics Workbench* and import the data:

> **File** | **Import High-throughput Sequencing Data** | **Fasta**

This will bring up the dialog shown in figure 19.8

Select the `ChIP-seq reads.fa` file and make sure the **Paired reads** checkbox is NOT checked. The option to discard read names is not significant in this context because of the relatively small amount of reads. Click **Next**, **Save** the imported reads list and click **Finish**.

After a short while, the 142,000 reads in the file have been imported. Next, import the reference genome sequence also included in the zip file:

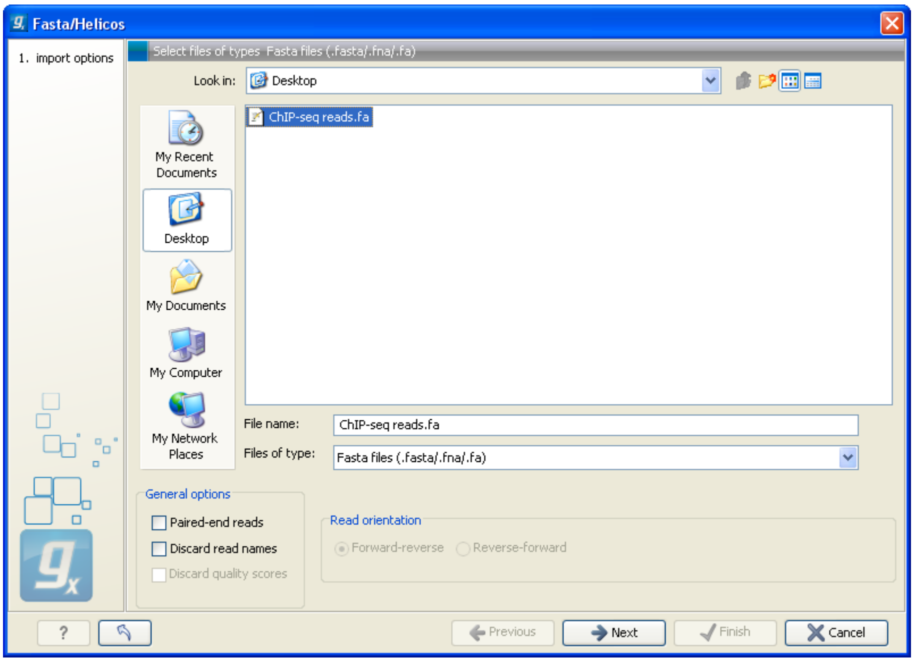


Figure 2.40: *When analyzing your own data, you would select the sequencing technology appropriate for your data. This data set consists of a fasta file, so you select Fasta.*

File | Import ( ) | Locate "NC_000021.gbk" | Select

Note that this is a genbank file imported using the "normal" import tool. Next-generation sequencing data needs to be imported using the special tool in the toolbox because they often have a more complex structure (in this case you could actually have used the normal import, because it is a simple fasta file).

2.8.2 Mapping the reads to the reference

First step in the analysis is to map the reads to the reference genome:

Toolbox | High-throughput Sequencing ( ) | Map Reads to Reference ( )

This shows the dialog in figure 2.41).

Select the `ChIP-seq reads` ( ) sequence list and add it to the panel to the right. Clicking **Next** will allow you to select a reference sequence as shown in figure 2.42.

At the top you select the `NC_000021` ( ) by clicking the **Browse and select element** ( ) button. You can select either single sequences or a list of sequences as reference sequences, but in this case just select this single chromosome.

Clicking **Next** to select the assembly settings as shown in figure 2.43.

For ChIP-seq, we recommend stringent mapping settings as shown in figure 2.43. Setting the limit to 2 and mismatch cost to 1, you allow one mismatch per read or 2 un-aligned nucleotides at the ends. Since this data set is artificial, the settings are not important for the result of this tutorial, but when you work with your own data, this is important. For more information about the other settings, please click the **Help** (?) button.

Click **Next** and you will see the dialog shown in figure 2.44.

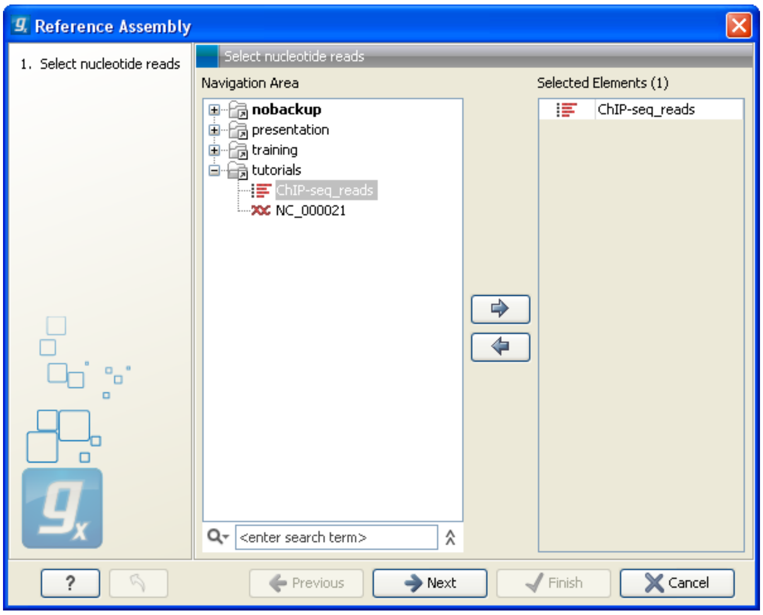


Figure 2.41: *Select sequence list containing the reads. The reference sequence will be selected in the next step.*

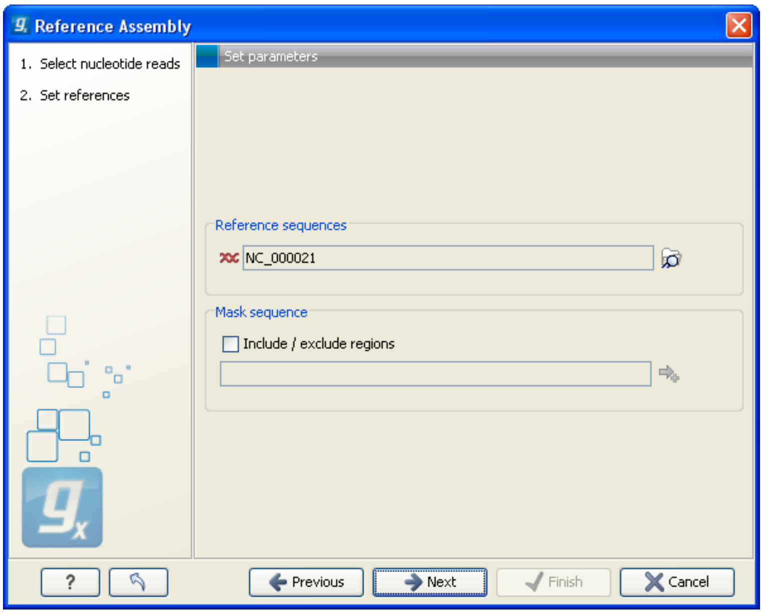


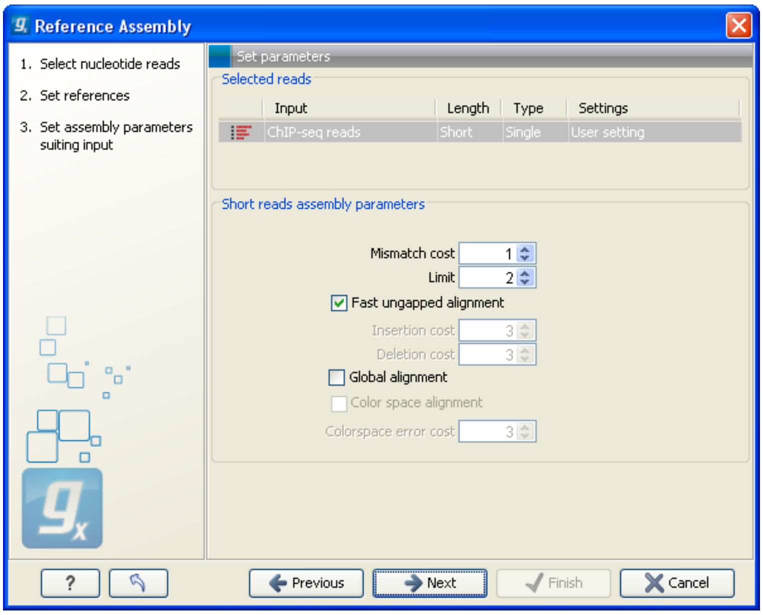
Figure 2.42: *Specifying the reference sequences and masking.*

In this dialog, select to ignore the non-specific matches. Click **Next** and **Finish**.

You can follow the progress of the mapping both in the status bar at the bottom left corner and under the **Processes** tab. There is also a log showing the progress. Because of the quite big reference sequence (Human chromosome 21, with a size of 47 Mbp), it takes a while to map[5].

Save (💾) the result into a folder in the **Navigation Area**.

[5] If you think it takes too long on your computer to finish, there is a smaller subset available at http://download.clcbio.com/testdata/raw_data/chip-seq-subset.zip



Selected reads

Input	Length	Type	Settings
ChIP-seq reads	Short	Single	User setting

Short reads assembly parameters

Mismatch cost 1

Limit 2

☑ Fast ungapped alignment

Insertion cost 3

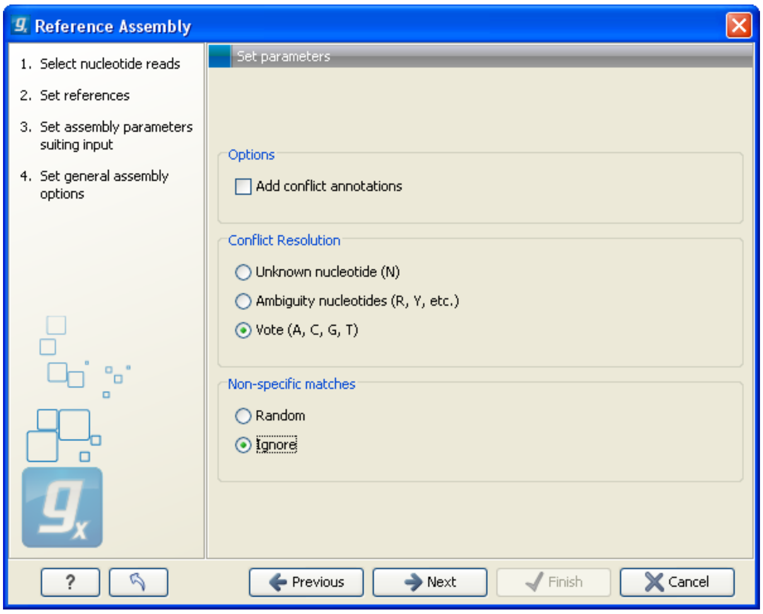
Deletion cost 3

□ Global alignment

□ Color space alignment

Colorspace error cost 3

Figure 2.43: *A stringent read matching is desired for ChIP-seq.*


Reference Assembly
1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input
4. Set general assembly options
Set parameters
Previous
Next
Finish
Cancel

Options

□ Add conflict annotations

Conflict Resolution

○ Unknown nucleotide (N)

○ Ambiguity nucleotides (R, Y, etc.)

● Vote (A, C, G, T)

Non-specific matches

○ Random

● Ignore

Figure 2.44: *Selecting to ignore non-specific matches.*

2.8.3 Running the ChIP sequencing analysis

The result of the read mapping is now used as input to the ChIP-seq function which surveys the pattern in coverage and read orientation to detect significant peaks:

Toolbox | High-throughput Sequencing ( ) | ChIP-Seq Analysis ( )

This opens a dialog where you select the `ChIP-Seq reads mapping` ( ) and click **Next**.

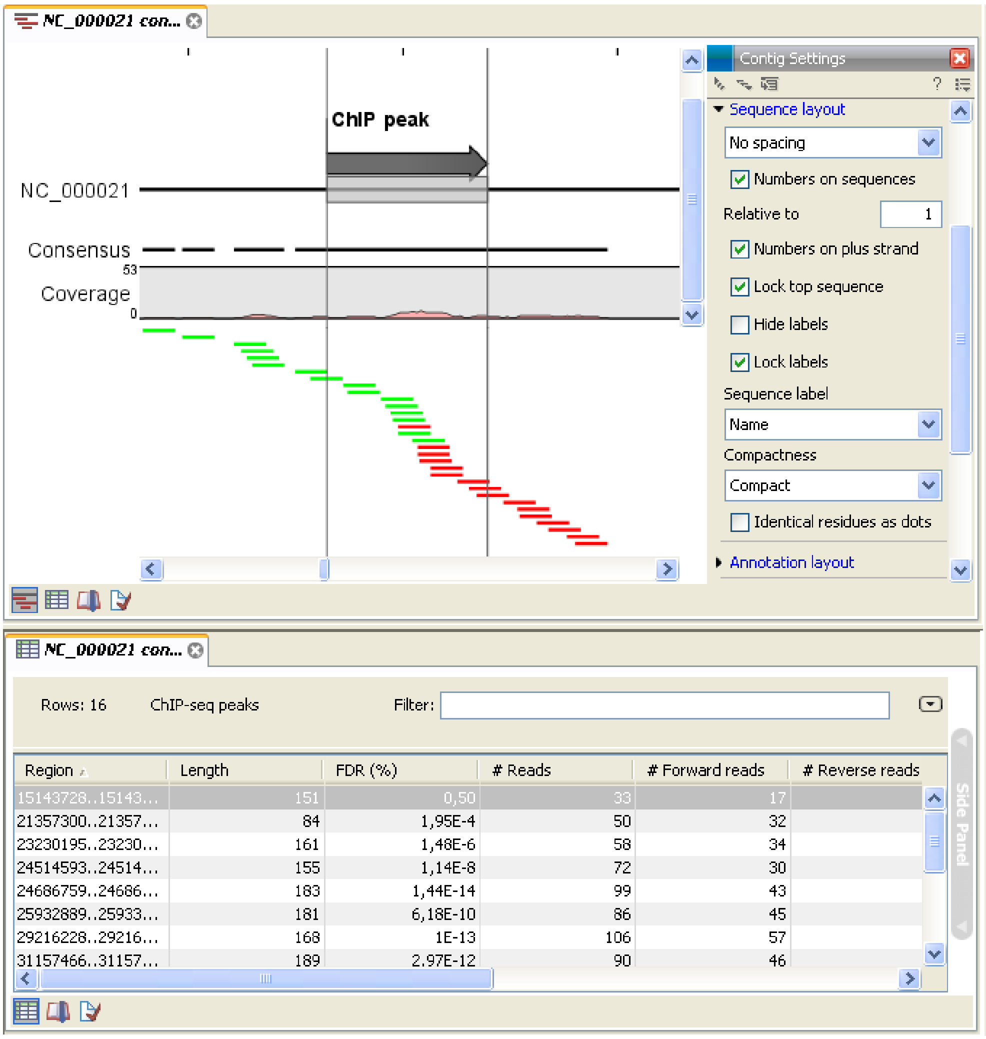
Make sure that all parameters are set to default and click **Next**. You can set the parameters to default by clicking the button ( ) at the bottom of the dialog. Repeat this procedure until you can click **Finish**.

Remember that you can get details of the ChIP-seq analysis from the user manual: Click the **Help**

(?) button to read this or read the advanced tutorial.

As a result of the analysis, annotations are added to the reference sequence of the mapping input file where significant peaks are detected, and a table is displayed below the mapping. Click the first row in the table in order to jump to the corresponding position on the reference. At this point you may want to adjust a few view settings to get a better overview: In the **Side Panel**, under **Sequence layout**, set **Compactness** to Compact. Next, click **Zoom out** ( ) in the Toolbar and click 6-8 times in the view.

Your screen should now look like figure 2.45.



No spacing

☑ Numbers on sequences

Relative to 1

☑ Numbers on plus strand

☑ Lock top sequence

□ Hide labels

☑ Lock labels

Sequence label: Name

Compactness: Compact

□ Identical residues as dots

Rows: 16 ChIP-seq peaks Filter:

Region	Length	FDR (%)	# Reads	# Forward reads	# Reverse reads
15143728..15143...	151	0,50	33	17	
21357300..21357...	84	1,95E-4	50	32	
23230195..23230...	161	1,48E-6	58	34	
24514593..24514...	155	1,14E-8	72	30	
24686759..24686...	183	1,44E-14	99	43	
25932889..25933...	181	6,18E-10	86	45	
29216228..29216...	168	1E-13	106	57	
31157466..31157...	189	2.97E-12	90	46	

Figure 2.45: *A split view showing the peak table and the mapping.*

Note the nicely distributed green (forward) and red (reverse) reads for this peak. You can browse through all the 16 peaks found for this sample by selecting in the table.

Since this is an artificial data set, we actually know what the results should be. If you go to the **Side Panel** and click **Annotation types**, you will find an annotation type called "Misc. binding". If you click that, the spiked-in peaks will be revealed on the reference sequence. You will see that all the peaks found by the *CLC Genomics Workbench* are covered by spike annotations (you can also see that in the table).

There are a few spiked in peaks that are not found since these only contain a very small number of reads - you can browse through them by clicking the small arrow ( ) next to the checkbox.

2.9 ChIP sequencing: Understanding the details

This tutorial takes you through some of the details of the ChIP-seq analysis in the *CLC Genomics Workbench*. We recommend having a look at the basic ChIP-seq tutorial first (see `http://www.clcbio.com/tutorials`), since we will skip part of the work flow here.

2.9.1 Data set

The data set used derives from a study reported in Nielsen et al., 2008. In order to make the data set comprehendible and make the computing time and requirements low, the original full data set has been reduced.

First of all, only one of the 18 samples have been used. It is the sample of *PPAR*γ on day 6. This sample has been mapped against the mouse refseq genome, and two regions of chromosome 7 have been taken out for use in this tutorial. The reference sequence used is 10 Mbp, and there are 23,600 reads of 32 bp each.

The import and mapping was covered in the basic tutorial, so we proceed directly to the ChIP-seq analysis.

2.9.2 Getting the right layout

First, **Import** ( ) the `chip-seq_pparg-subset.zip` file that you can download from `http://download.clcbio.com/testdata/raw_data/chip-seq_pparg-subset.zip`. Now, open the **ChIP-sequencing analysis** ( ) dialog from the **Toolbox**, select the mapping result and click **Next**.

Make sure that all parameters are set to default. You can set the parameters to default by clicking the button ( ) at the bottom of the dialog. Uncheck the **Shift reads based on fragment length** setting at this step.

Click **Next** and set parameters to default, click **Next** and set parameters to default, deselect the **Make log** checkbox, and click **Finish**. We will come back to learn more about the parameters later on. With these settings, you should be able to detect 12 peaks.

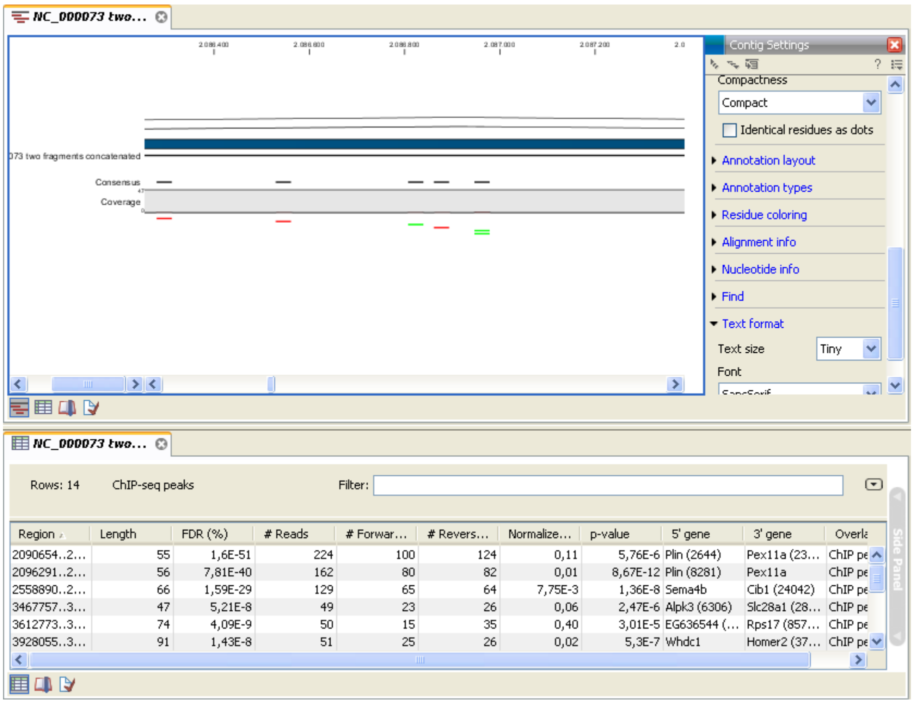
When the result is opened, you need to do a few customizations to make it better suited for interpretation. In the **Side Panel**, under **Sequence layout**, set **Compactness** to Compact. Below, under **Text format**, set the font size to small or tiny.

Next, click **Zoom out** ( ) in the Toolbar and click 6-8 times in the view.

Your screen should now look like figure 2.46.

2.9.3 Looking for known genes

The paper comments on the gene *perilipin (Plin)*, so we will now take a look at the binding sites surrounding that gene. In the peak table, simply type `Plin` into the **Filter** textbox, and you will



Compactness: Compact

□ Identical residues as dots

Text size: Tiny

Rows: 14 ChIP-seq peaks Filter:

Region	Length	FDR (%)	# Reads	# Forwar...	# Revers...	Normalize...	p-value	5' gene	3' gene	Overla
2090654..2...	55	1,6E-51	224	100	124	0,11	5,76E-6	Plin (2644)	Pex11a (23...	ChIP pe
2096291..2...	56	7,81E-40	162	80	82	0,01	8,67E-12	Plin (8281)	Pex11a	ChIP pe
2558890..2...	66	1,59E-29	129	65	64	7,75E-3	1,36E-8	Sema4b	Cib1 (24042)	ChIP pe
3467757..3...	47	5,21E-8	49	23	26	0,06	2,47E-6	Alpk3 (6306)	Slc28a1 (28...	ChIP pe
3612773..3...	74	4,09E-9	50	15	35	0,40	3,01E-5	EG636544 (...	Rps17 (857...	ChIP pe
3928055..3...	91	1,43E-8	51	25	26	0,02	5,3E-7	Whdc1	Homer2 (37...	ChIP pe

Figure 2.46: *The view has now been set up.*

now see one row in the table representing one peak located in an intron of *Pex11a* which sits next to *Plin*. Click the row to have the mapping view jump to see the peak.

You can now inspect the peak and if you zoom out you will see the adjacent *Plin* and *Pex11a* genes. Note that there is also two peaks between the two genes that is not found by the ChIP-sequencing tool. We will get back to that later.

Another gene commented in the paper is *Pnpla2* gene. Type this into the filter, and you will see two rows. The filter is very simple - it searches all the information in the table for the text you have input. Since the *Pnpla2* gene is represented in the columns for nearest gene 5' for one peak and as the nearest 3' gene in another peak, these two records are shown.

2.9.4 Going into detail with the parameters

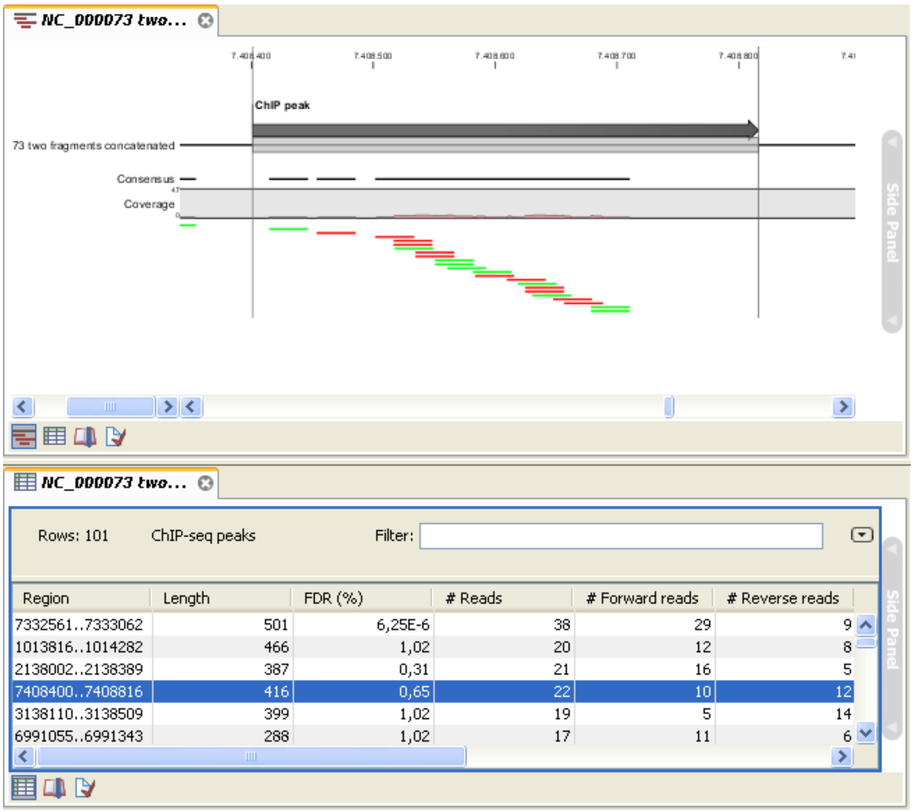
Next, we will go through some of the settings for the ChIP-seq analysis. First, click the tab of the mapping in the upper view, and click **Undo** (↶). This removes the peak annotations of the mapping.

Run the analysis again, but this time make sure to uncheck all check-boxes on step 3 except the **Boundary refinement**. All other options remain the same.

The resulting table should report 79 peaks.

Now sort the table on the **Normalized difference** column. The one with the highest difference has 2 forward reads and 13 reverse reads. Usually, a peak like that would not be trusted because you would expect roughly the same number of forward and reverse reads. When you run the ChIP-sequencing tool, you can enter a maximum limit for this value. The default is 0.4. You can see in the table that applying this setting would have excluded 14 peaks.

Next, sort the table on p-value. If you click the peak with a p-value of 0.6, you will see that the distribution of the reads is different from the other peaks we have seen (see figure 2.47).



Rows: 101 ChIP-seq peaks Filter:

Region	Length	FDR (%)	# Reads	# Forward reads	# Reverse reads
7332561..7333062	501	6,25E-6	38	29	9
1013816..1014282	466	1,02	20	12	8
2138002..2138389	387	0,31	21	16	5
7408400..7408816	416	0,65	22	10	12
3138110..3138509	399	1,02	19	5	14
6991055..6991343	288	1,02	17	11	6

Figure 2.47: *A peak with a random distribution of forward and reverse reads leading to a high p-value.*

When the fragments from the ChIP are sequenced, they are sequenced from the ends which means that you would expect forward reads upstream of the binding site and reverse reads downstream. This peak has a very random distribution of the reads and that is the reason behind the high p-value.

When we did the first round of analysis, we looked at the *Plin* gene and saw a peak in coverage that was not detected. Now that we have relaxed the parameters, we can go back and inspect the peaks again. Type in `Plin` in the filter text box. You can see that the peak just next to the start site of *Plin* is also detected. It has a p-value of 0.0008 and is thus above the default threshold of 0.0001 used in the first round of analysis.

We recommend running the ChIP-sequencing analysis several times using different parameters to get an idea of the best setting. If you start by running an analysis with relaxed parameters (e.g. not using the Normalized difference or the distribution P-value) you will be able to do a visual inspection of the peaks that would otherwise have been excluded and by interpretation narrow down the right parameter settings.

2.9.5 Extracting the peak regions

Once you have decided on the right settings for the analysis, make sure that all existing peak annotations are removed (right-click and delete annotations of type binding site) before you run the final analysis. In this way, the annotations on the sequences are right.

If you want to extract the sequence of the binding sites to do motif discovery, you can do this based on the binding site annotation. First, you need to download a plug-in to the Workbench called **Extract annotations**. You need to be connected to the internet to do this:

Help Menu Bar | Plug-ins and Resources... ( )

Click the **Download Plug-ins** tab. This will display an overview of the plug-ins that are available for download. Find the **Extract annotations** plug-in and click **Download and Install**. A dialog displaying progress is now shown, and the plug-in is downloaded and installed.

When you close the dialog, you will be asked whether you wish to restart the *CLC Genomics Workbench*. The plug-in will not be ready for use before you have restarted.

When you have re-started you can now extract all the parts of your genome that are covered by a "Binding site" annotation. First, right-click the name of the reference sequence in the mapping view and click **Open This Sequence**. This is because annotations can only be extracted from a sequence alone - not a mapping view.

Toolbox | General Sequence Analyses ( ) | Extract Annotations ( )

Click through using the default settings. You will now see a list of all the binding sites which you can e.g. **Export** ( ) in fasta format for use in a motif discovery tool.

2.10 RNA-Seq analysis part I: Getting started

This tutorial is the first part of a series of tutorials about RNA-Seq. The aim of the tutorials is to take you from start to end of an RNA-Seq analysis including mapping of reads, interpreting results, checking quality and finally doing statistical analysis. Along the way, we will focus on illustrating the effect of the parameters and choices made during the analysis.

The data used is from a study reported in [Mortazavi et al., 2008]. The data set consists of RNA-Seq data from three types of Mouse tissue: Brain, Liver and Skeletal muscle. Each of the tissues has been sampled twice, so there are 6 samples all in all.

2.10.1 Downloading and importing the data

At `http://www.clcbio.com/ngsexampledata` you find the following data:

Subset of the full data set This file can be imported using the standard import and includes a subset of the full data set including a region of chromosome 16 for use as a reference. When running the full data set, we extracted all the reads that matched the genes of this part of chromosome 16. Download and import this data set (using the normal import) for use in these tutorials.

Experiments with the full data set Later on, we will work on experiments generated from the full data set. Download and import this data set (using the normal import) for use in these tutorials.

Once downloaded and imported, you should have the following folders and data in the **Navigation Area** (see figure 2.48).

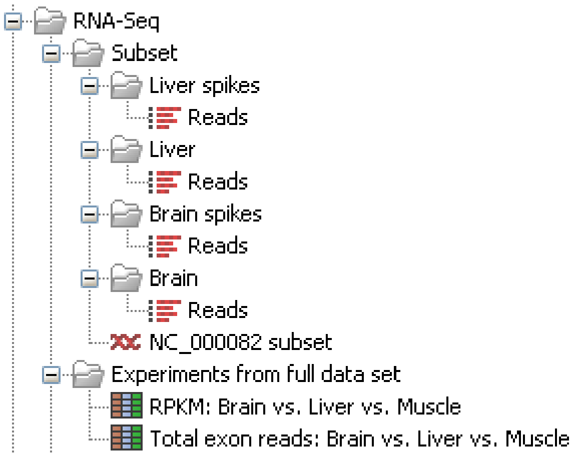

RNA-Seq
Subset
Liver spikes
Reads
Liver
Reads
Brain spikes
Reads
Brain
Reads
NC_000082 subset
Experiments from full data set
RPKM: Brain vs. Liver vs. Muscle
Total exon reads: Brain vs. Liver vs. Muscle

Figure 2.48: *The subset of the full data set has been imported together with the experiments generated from the full data set.*

2.10.2 Running the RNA-Seq analysis

Now, you can start the actual analysis. First step is to transform the list of reads into what we call an RNA-Seq sample which is basically a list of genes with expression values. To do this, go to:

Toolbox | High-throughput Sequencing ( ) | RNA-Seq Analysis ( )

This opens a dialog where you select the sequencing reads from the *Brain spike* sample, as shown in figure 2.49.

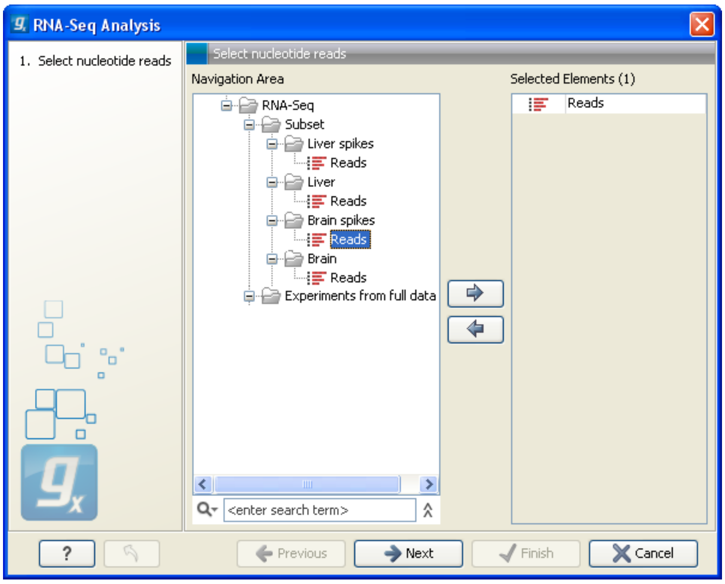
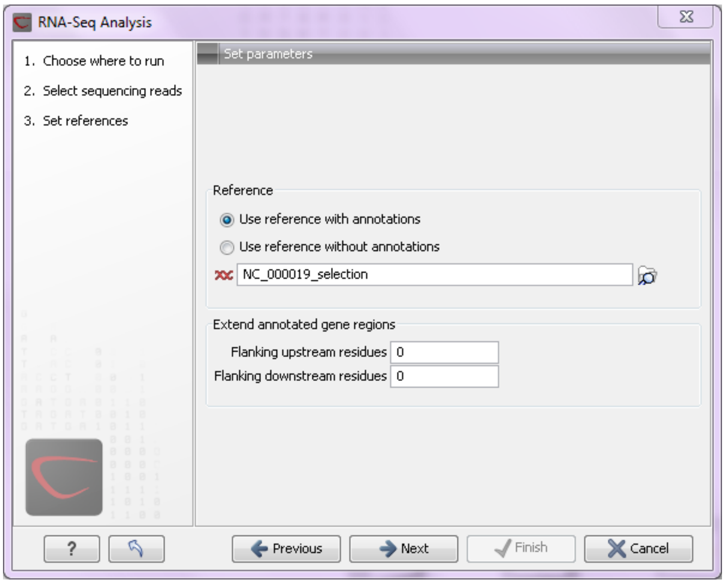


Figure 2.49: *Selecting the Brain spikes sample for RNA-Seq analysis.*

Click **Next** when the data is listed in the right-hand side of the dialog.

You are now presented with the dialog shown in figure 2.50.

Since we are using (part of) the ref-seq annotated mouse genome, choose **Use reference with annotations**. Click ( ) to select the reference sequence *NC_000082 subset*.



RNA-Seq Analysis

1. Choose where to run
2. Select sequencing reads
3. Set references

Set parameters

Reference

● Use reference with annotations

○ Use reference without annotations

NC_000019_selection

Extend annotated gene regions

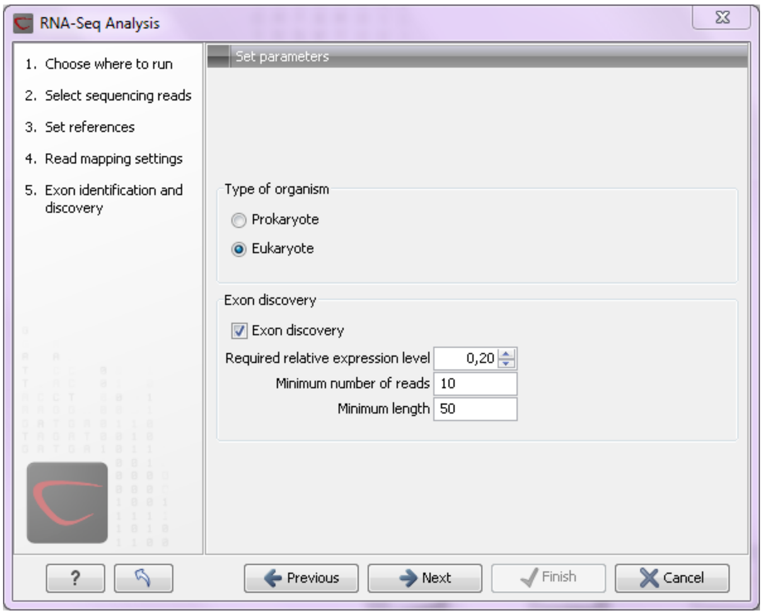
Flanking upstream residues: 0

Flanking downstream residues: 0

Figure 2.50: *Choosing the annotated reference sequence.*

Click **Next** where you can set parameters for the mapping. Leave these settings at their default - we will focus on these later on. (You can set the parameters to default by clicking the button (↶) at the bottom of the dialog, but then you will have to define the reference sequence again).

Clicking **Next** will show the dialog in figure 2.51.



RNA-Seq Analysis

1. Choose where to run
2. Select sequencing reads
3. Set references
4. Read mapping settings
5. Exon identification and discovery

Set parameters

Type of organism

○ Prokaryote

● Eukaryote

Exon discovery

☑ Exon discovery

Required relative expression level: 0,20

Minimum number of reads: 10

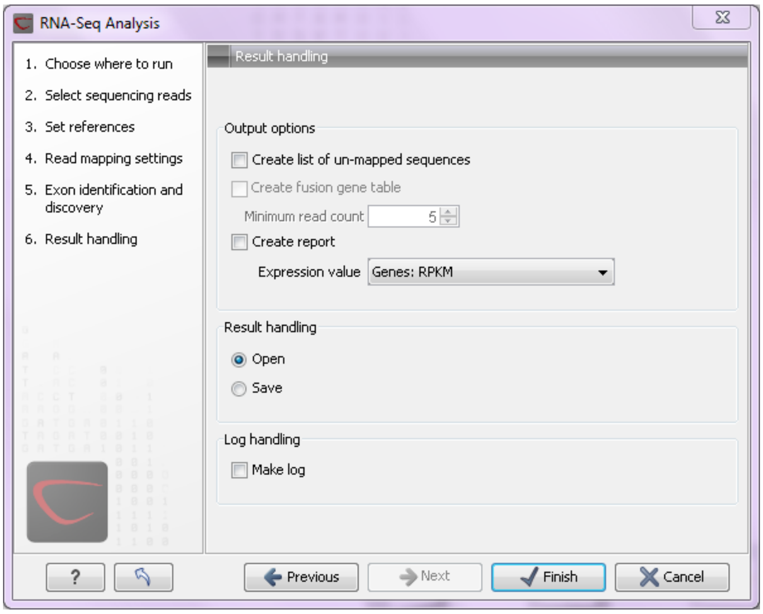
Minimum length: 50

Figure 2.51: *Exon discovery.*

The choice between **Prokaryote** and **Eukaryote** is basically a matter of telling the Workbench whether you have introns in your reference. In order to select **Eukaryote**, you need to have reference sequences with annotations of the type mRNA (this is the way the Workbench expects exons to be defined). The reference sequence provided with this tutorial includes mRNA annotations (they are the green annotations), so you select **Eukaryote** in this wizard.

Below you can specify settings for discovering novel exons. We will investigate this in detail later on.

Clicking **Next** will allow you to specify the output options as shown in figure 2.52.



RNA-Seq Analysis

1. Choose where to run
2. Select sequencing reads
3. Set references
4. Read mapping settings
5. Exon identification and discovery
6. Result handling

Result handling

Output options

□ Create list of un-mapped sequences

□ Create fusion gene table

Minimum read count 5

□ Create report

Expression value Genes: RPKM

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 2.52: *Selecting the output of the RNA-Seq analysis.*

Uncheck the **Create list of un-mapped reads**, **Create report** and **Make log** and click **Finish**.

The standard output is a table showing mapping statistics on each gene.

2.10.3 Interpreting the brain spikes analysis result

The result of the RNA-Seq analysis is shown in figure 2.53.

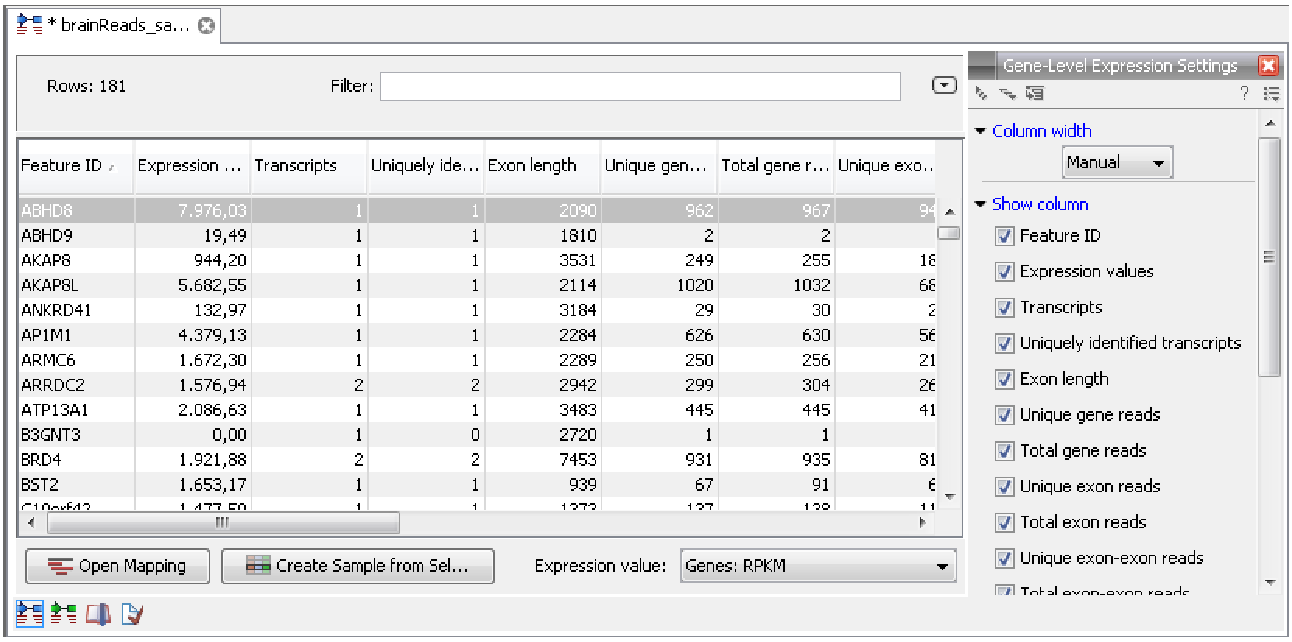
The **Expression values** column is per default based on the RPKM. Change the measure to use **Total exon reads** instead by clicking at the bottom of the view (we will go into more details with expression measures in part II). Now sort the table on the new expression value by clicking the column header twice. Find the *Ahsg* gene (4th from the top of the list) and double-click.

When the result is open, you need to do a few customizations to make the view better suited for interpretation. In the **Side Panel**, under **Sequence layout**, set **Compactness** to Compact. Below, under **Text format**, set the font size to small or tiny. To save these customizations so that they take effect next time you open a mapping, click the **Save/Restore Settings** button ( ) at the top of the **Side Panel** and click **Save Settings**. Give your settings a name and make sure the check box to **Always apply these settings** is checked.

Double-click the tab of the view (or press Ctrl + M) to **Maximize the view** and click **Fit Width** ( ) in the tool bar to zoom out to see the full gene. You should now have a view similar to figure 2.54.

You can now see distinct peaks of coverage below the exons which are marked in green. Scroll slowly down on the scroll bar at the right hand side of the view. You will begin to see reads that have been mapped across exon-exon boundaries.

Click **Zoom in** ( ) and click-and-drag a rectangle around one of the exons. In this way you can



* brainReads_sa...

Rows: 181 Filter:

Feature ID	Expression ...	Transcripts	Uniquely ide...	Exon length	Unique gen...	Total gene r...	Unique exo...
ABHD8	7.976,03	1	1	2090	962	967	94
ABHD9	19,49	1	1	1810	2	2	
AKAP8	944,20	1	1	3531	249	255	18
AKAP8L	5.682,55	1	1	2114	1020	1032	68
ANKRD41	132,97	1	1	3184	29	30	2
AP1M1	4.379,13	1	1	2284	626	630	56
ARMC6	1.672,30	1	1	2289	250	256	21
ARRDC2	1.576,94	2	2	2942	299	304	26
ATP13A1	2.086,63	1	1	3483	445	445	41
B3GNT3	0,00	1	0	2720	1	1	
BRD4	1.921,88	2	2	7453	931	935	81
BST2	1.653,17	1	1	939	67	91	6

Open Mapping | Create Sample from Sel... | Expression value: Genes: RPKM

Gene-Level Expression Settings

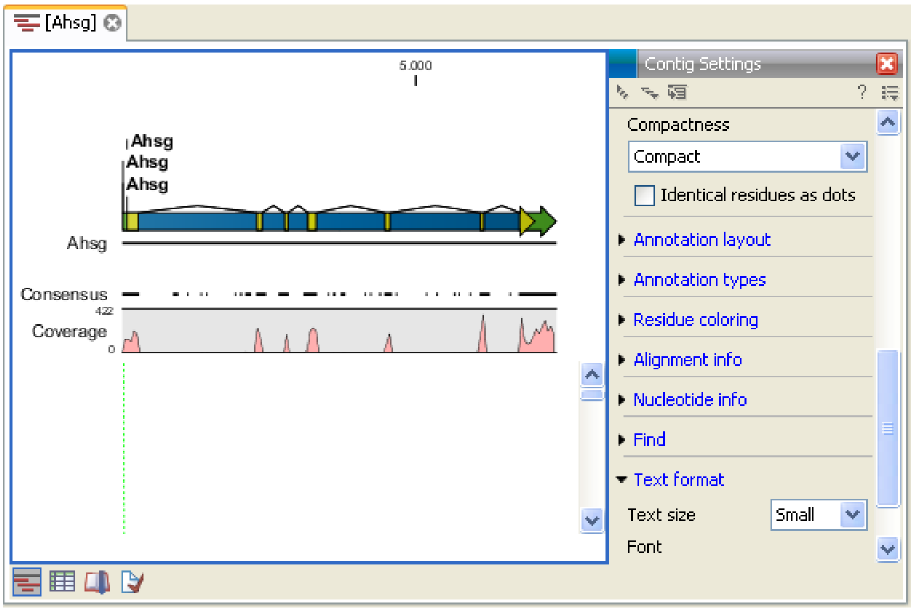
Column width

Manual

Show column

☑ Feature ID
☑ Expression values
☑ Transcripts
☑ Uniquely identified transcripts
☑ Exon length
☑ Unique gene reads
☑ Total gene reads
☑ Unique exon reads
☑ Total exon reads
☑ Unique exon-exon reads
☑ Total exon-exon reads

Figure 2.53: *A table with expression values for all genes.*



Contig Settings

Compactness

Compact

□ Identical residues as dots

Annotation layout
Annotation types
Residue coloring
Alignment info
Nucleotide info
Find
Text format

Text size Small

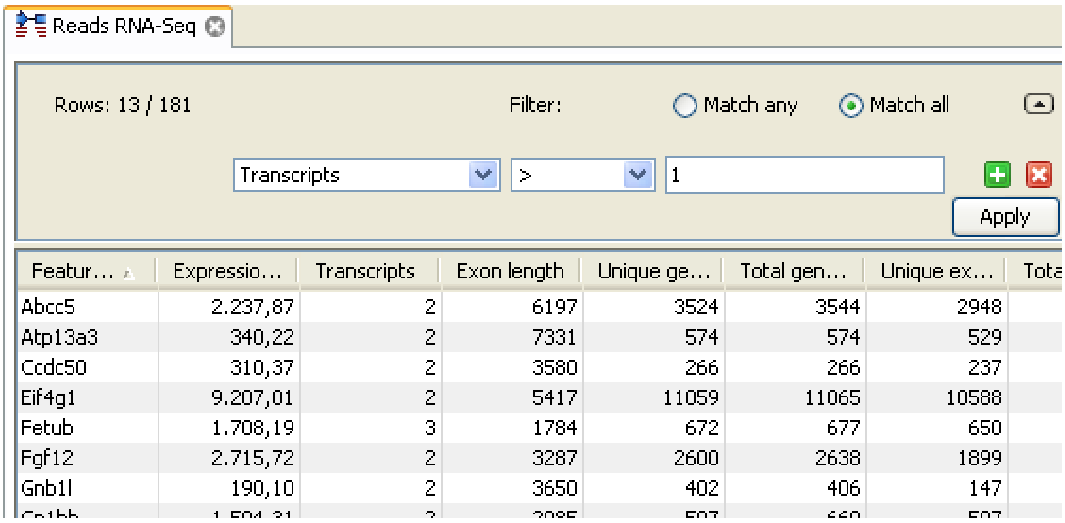
Font

Figure 2.54: *The reads mapped to the Ashg gene.*

zoom in to see more details of a particular exon. If you zoom all the way in, you will be able to see the nucleotide level and the alignment of the reads (to see the actual nucleotide level of the reads, you need to set the **Compactness** settings back to **Not compact**).

Close the view and go back to the RNA-seq sample. In the 'Transcripts' column you can see that the *Ahsg* gene only has one transcript annotated. Use the **Advanced filter** (▾) at the upper right hand part of the RNA-seq sample table view) to identify genes with more than one transcript annotated (set the filter to `Transcripts > 1` and press **Apply** as shown in figure 2.55).

The *Fetub* gene has three transcripts annotated. Open the mapping for this gene and press **Fit**



Reads RNA-Seq

Rows: 13 / 181

Filter: ○ Match any ● Match all

Transcripts | > | 1

Apply

Featur...	Expressio...	Transcripts	Exon length	Unique ge...	Total gen...	Unique ex...	Tota
Abcc5	2.237,87	2	6197	3524	3544	2948	
Atp13a3	340,22	2	7331	574	574	529	
Ccdc50	310,37	2	3580	266	266	237	
Eif4g1	9.207,01	2	5417	11059	11065	10588	
Fetub	1.708,19	3	1784	672	677	650	
Fgf12	2.715,72	2	3287	2600	2638	1899	
Gnb1l	190,10	2	3650	402	406	147	

Figure 2.55: *Using the advanced filter to only show genes with more than one annotated transcript.*

width ( ) to zoom out completely and get an overview of the mapping to this gene.

One of the three transcripts annotated for *Fetub* uses a different first exon from the other two transcripts. There is no coverage in this exon at all, and thus no evidence for expression of the alternative first exon isoform. The other two transcripts have the same first exon but one skips the second exon of the other. If you scroll down you see both reads that span from exon 2 to exon 3 and reads that span from exon 2 to exon 4. Thus, there is evidence for both of these splice variants (see figure 2.56).

Close the view and you are ready for part II: Non-specific matches and expression values.

2.11 RNA-Seq analysis part II: Non-specific matches and expression measures

This tutorial is the second part of a series of tutorials about RNA-Seq analysis. We continue working with the data set introduced in the first tutorial.

In this tutorial we will first explain how non-specific matches are treated, and second we will explain the effect of using different expression measures.

2.11.1 Running the same data set with and without non-specific matches

Imagine a situation where you have nine reads that match equally well on two different genes. Since it is not possible to tell which transcript the reads actually came from, the Workbench has to decide where to place them. Based on other reads that are matched *uniquely* to the genes, the Workbench estimates the expression of each gene and use that as a weight to distribute the reads. In a situation where one of the genes have twice the number of unique matches, it will on average receive six of the nine reads whereas the other one would get three.

Now, we will show the effect of including these non-specific matches in the analysis. In the analysis of the first tutorial, the **Maximum number of hits for a read** was set to 10. This means that all reads that match in 10 or fewer places will be included in the mapping, with multi-hitting reads being distributed as described above. Now, run a new **RNA-Seq Analysis ( )** on the *Brain*

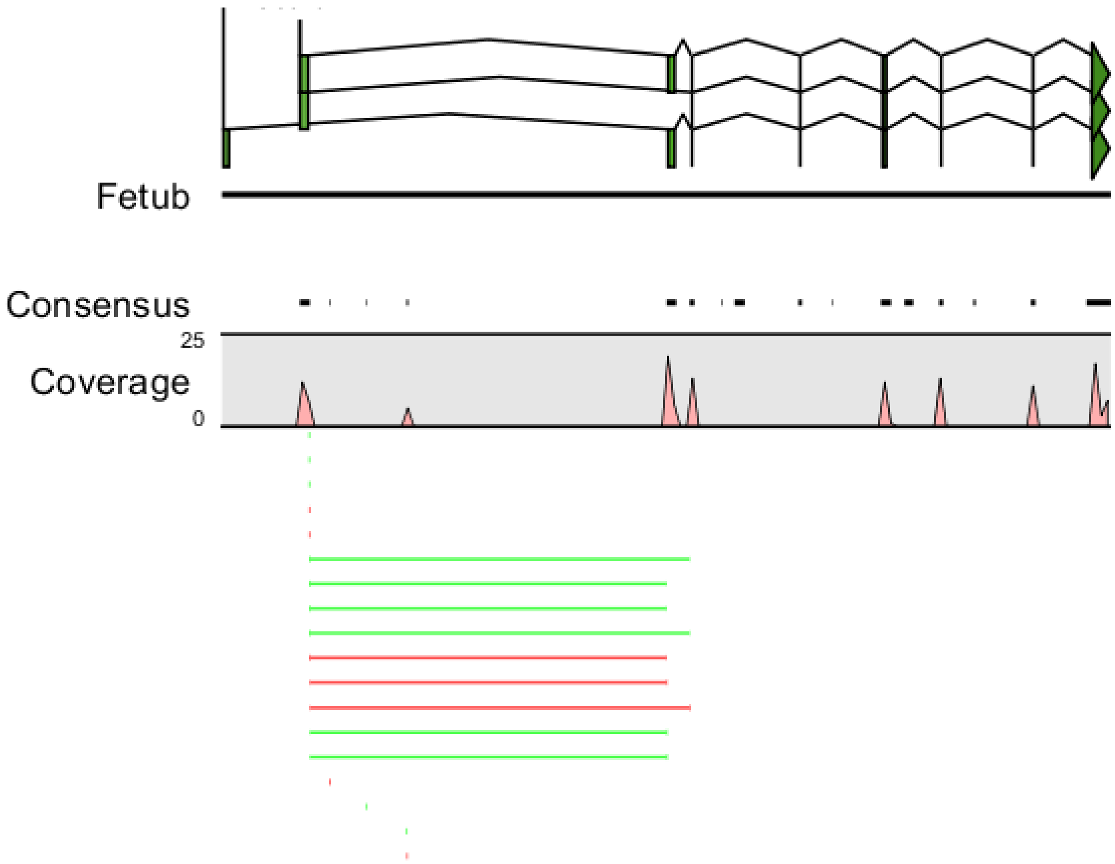


Figure 2.56: *Reads showing evidence for expression of two isoforms.*

spike sample, but this time set the **Maximum number of hits for a read** to 1. You find this setting in step 2 - leave the rest of the settings as they are.

You will now end up with an RNA-Seq sample where all reads matching in more than one position are excluded. If you go to the **History** () view of this new sample, you can see how many reads were not mapped. If you compare these numbers, you can see the first sample has 214631 unmapped reads whereas the second run without multihit reads has 226574 unmapped reads (figure 2.57).

Save () the two samples with meaningful names, e.g. *including non-specific* and *only specific*.

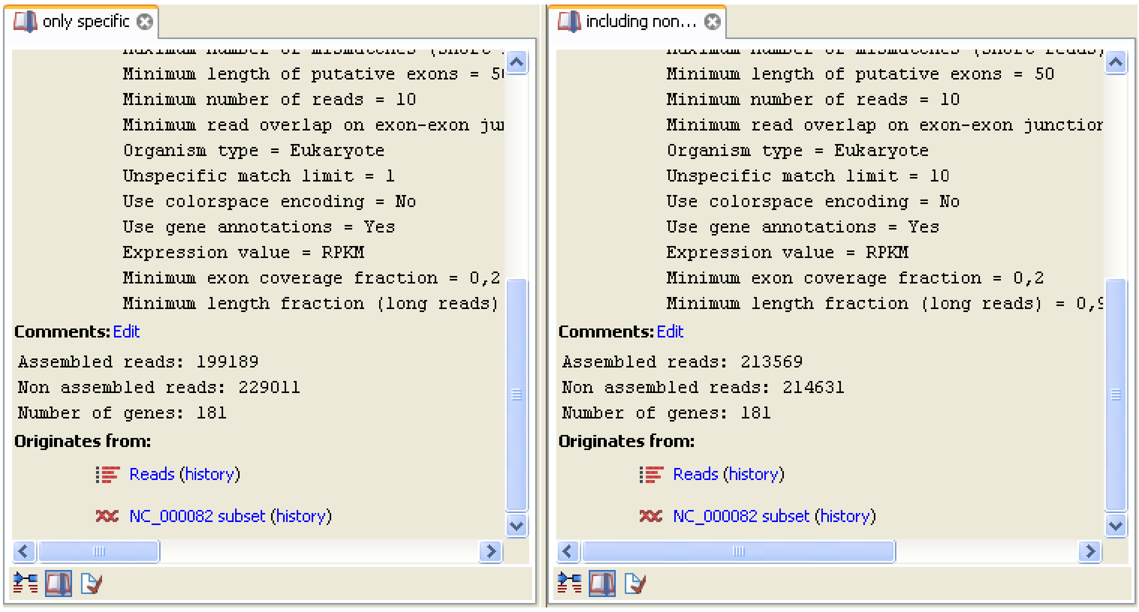
2.11.2 Comparing the data in a scatter plot

Now, we want to see what this difference means in terms of the expression values. In order to compare the two samples, we set up an experiment:

Toolbox | **Expression Analysis** () | **Set Up Experiment** ()

Select the two RNA-Seq samples () that you have just saved and click **Next**. Choose an un-paired, two-group experiment, set the **Value to use in experiment** to to **Total exon reads** and click **Next**. Name the groups *including non-specific* and *only specific* and click **Next**. Right-click each of the samples and assign it to the appropriate group. Click **Finish**.

You will now have an experiment based on the two samples. We will go into more details with



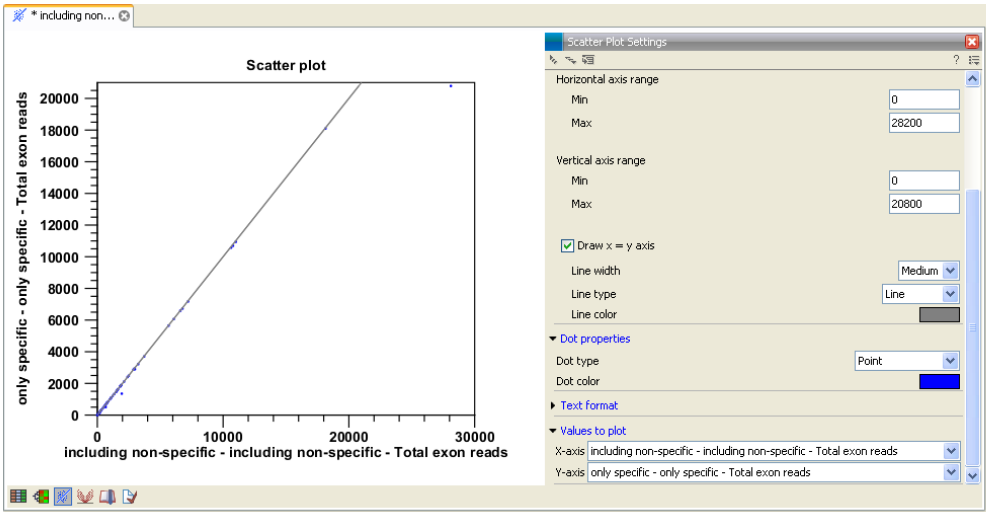
```
        Maximum number of mismatches (short r
        Minimum length of putative exons = 5
        Minimum number of reads = 10
        Minimum read overlap on exon-exon ju
        Organism type = Eukaryote
        Unspecific match limit = 1
        Use colorspace encoding = No
        Use gene annotations = Yes
        Expression value = RPKM
        Minimum exon coverage fraction = 0,2
        Minimum length fraction (long reads)
Comments: Edit
Assembled reads: 199189
Non assembled reads: 229011
Number of genes: 181
Originates from:
        Reads (history)
        NC_000082 subset (history)
```

```
        Maximum number of mismatches (short reads)
        Minimum length of putative exons = 50
        Minimum number of reads = 10
        Minimum read overlap on exon-exon junctior
        Organism type = Eukaryote
        Unspecific match limit = 10
        Use colorspace encoding = No
        Use gene annotations = Yes
        Expression value = RPKM
        Minimum exon coverage fraction = 0,2
        Minimum length fraction (long reads) = 0,9
Comments: Edit
Assembled reads: 213569
Non assembled reads: 214631
Number of genes: 181
Originates from:
        Reads (history)
        NC_000082 subset (history)
```

Figure 2.57: *Comparing the history entries for the two samples.*

the experiment later - for now we are interested in looking at the scatter plot. Click the **Scatter plot** ( ) icon at the bottom of the view.

At the bottom of the **Side Panel** you select the values to plot. Select *including non-specific Total exon reads* versus *only specific Total exon reads*, and you will see a view as shown in figure 2.58.



Horizontal axis range
Min: 0
Max: 28200

Vertical axis range
Min: 0
Max: 20800

☑ Draw x = y axis
Line width: Medium
Line type: Line
Line color

Dot type: Point
Dot color

X-axis: including non-specific - including non-specific - Total exon reads
Y-axis: only specific - only specific - Total exon reads

Figure 2.58: *A scatter plot showing the effect of including non-specific matches in the expression measure.*

The scatter plot now shows the expression levels of the two samples. Since the RNA-Seq analysis was run on the same data set with the only difference being the treatment of non-specific matches, you can now see the direct effect of using and distributing the non-specific matches in this way.

You can now see that many of the genes have close to identical expression measures (they are located along the x=y line in the plot), but there are some that show higher expression in the sample including non-specific matches. To see the outliers more clearly, set the **Dot type** under **Dot properties** in the **Side Panel** to **Dot**.

The most outlying gene is *Sept5*. If you place your mouse on the dot as shown in figure 2.59, you can see the feature ID (gene name) and the x and y values of the dot.

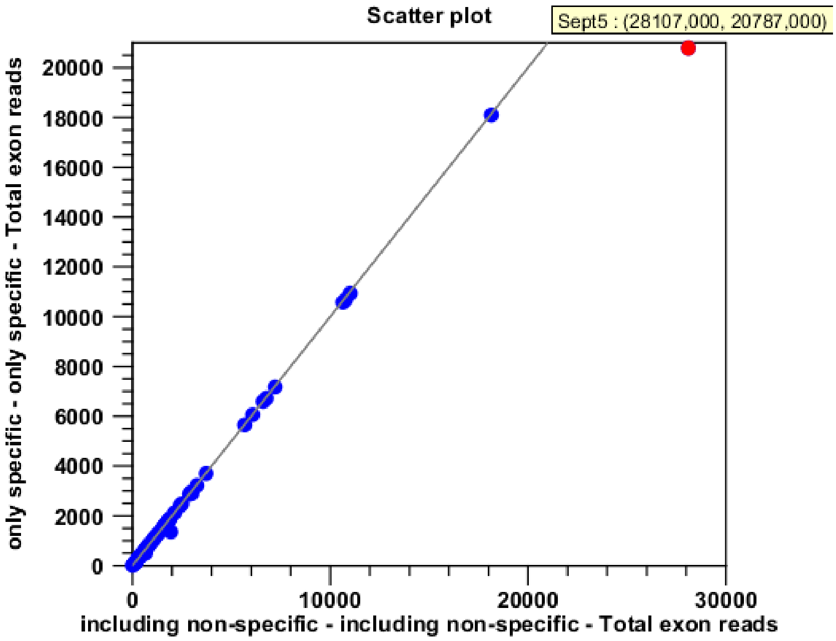


Figure 2.59: *Gene Sept5 is one of the genes showing a notable difference in expression.*

Open the *including non-specific* RNA-Seq sample and locate this gene by typing *Sept5* in the filter at the top. Double-click to open the mapping. When you zoom out ( ) and scroll along to the end of the gene, you will see a lot of reads that are yellow. Yellow is the color used for non-specific reads. In this case, all these yellow reads are the ones contributing to a higher expression measure when you compare in the scatter plot.

By looking at the gene annotations, you can also see the reason why there are so many non-specific matches. As shown in figure 2.60, there is an overlapping gene near the end. This means that all the reads that map to this part of the *Sept5* gene also map equally well to the beginning of the *Gp1bb* gene. These reads are then treated as non-specific matches.

If you opened the *Gp1bb* mapping you would also see the non-specific reads at the beginning where it overlaps with **Sept5**. Because we can see the overlap, we know why we have non-specific matches, but it could be that these reads would also match other places on the reference. It's easy to check if the same region is present other places in the reference by conducting a BLAST search:

> **select the relevant part of the gene|right-click the selection|BLAST against Local Data ( )**

This is illustrated in figure 2.61

Close the BLAST dialog again since we are not going into details with BLAST search in this tutorial. Please read the BLAST tutorial to learn more about BLAST in *CLC Genomics Workbench*. The idea with BLASTing this selection would be to use the reference sequence as target and see how many hits you would find. In this case there is only one good hit, but if you have a region of

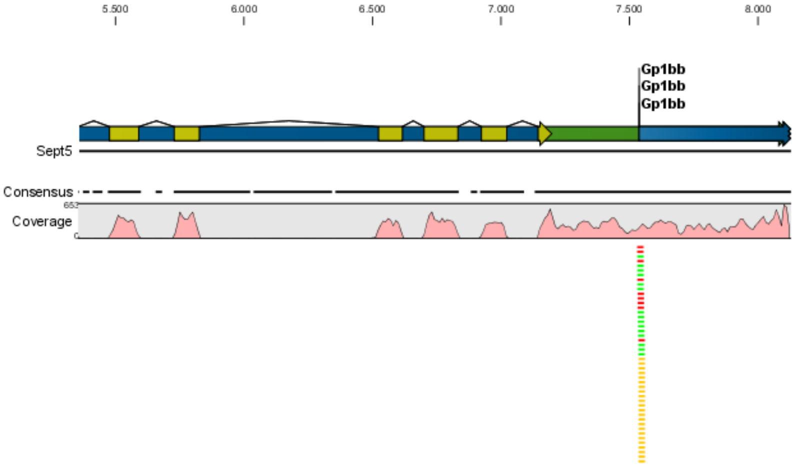


Figure 2.60: *Gp1bb is overlapping the end of the Sept5 gene.*

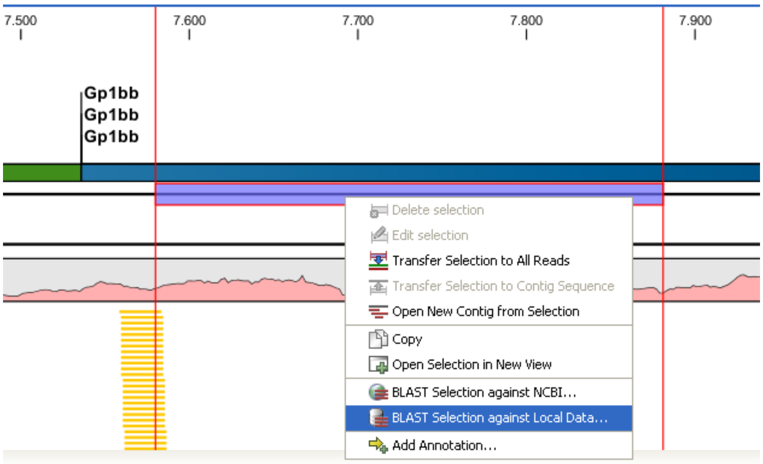

7.500
7.600
7.700
7.800
7.900
Gp1bb
Gp1bb
Gp1bb
Delete selection
Edit selection
Transfer Selection to All Reads
Transfer Selection to Contig Sequence
Open New Contig from Selection
Copy
Open Selection in New View
BLAST Selection against NCBI...
BLAST Selection against Local Data...
Add Annotation...

Figure 2.61: *BLAST against the reference.*

non-specific matches that are not due to overlapping genes, you can use this approach to try to identify which other gene is "competing" for these reads.

Close (☒) the mapping view, go back to the experiment and switch to the table view (▦). Enter *Gp1bb* in the filter and click with your mouse on the *Gp1bb* gene. Switch back to the scatter plot ( ) and *Gp1bb* will now be high-lighted with a red color. Click **Zoom in** ( ) and click a couple of times on the gene to zoom in (see figure 2.62).

You can now see that this gene also exhibits differential expression between these two samples, but to a lesser degree than the *Sept5* gene. Open the *Gp1bb* mapping from the *including non-specific* sample, and you can see that there are fewer yellow reads than in the *Sept5* mapping. As explained above, the non-specific reads are distributed according to the number of unique reads, and when you compare the two results, it is evident that there are many more unique reads in the *Sept5* gene (you can easily see the difference in the RNA-Seq result table as shown in figure 2.63).

From the scatter plot in figure 2.58, it is obvious that the decision on whether to include non-specific matches or not is very important. For some genes, the difference in expression is highly significant. This trend becomes even more evident when looking at the full data set where the proportion of non-specific matches is even higher (with the full reference transcriptome, there

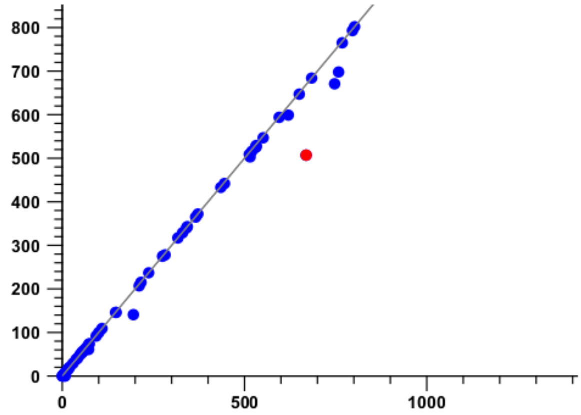
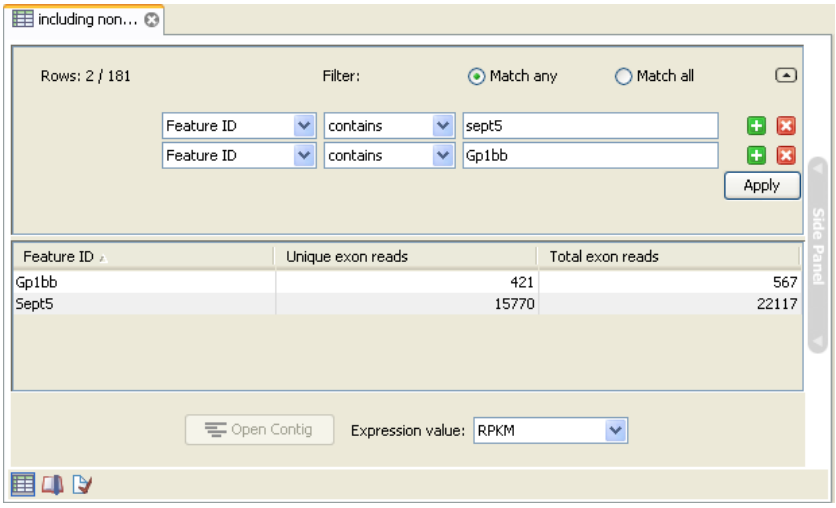


Figure 2.62: *Zooming in on the Gp1bb gene in the scatter plot.*



Rows: 2 / 181 Filter: ● Match any ○ Match all

Feature ID	contains	sept5
Feature ID	contains	Gp1bb

Feature ID	Unique exon reads	Total exon reads
Gp1bb	421	567
Sept5	15770	22117

Expression value: RPKM

Figure 2.63: *Comparing the number of unique reads between Gp1bb and Sept5.*

is a greater chance of finding sequences that are represented more times, e.g. arising through gene duplications).

It is hard to make general recommendations on how to treat non-specific matches. One of the pitfalls when including non-specific matches is that the number of unique matches can be too low to ensure a reliable distribution of the non-specific matches. One way of approaching this problem would be to run the same data set with different settings as shown in this tutorial. That will enable you to perform random checks of the genes whose expression is significantly altered, and you will be able to identify this kind of pattern. On the other hand, if you completely disregard non-specific reads, you may underestimate the expression levels of genes in gene families.

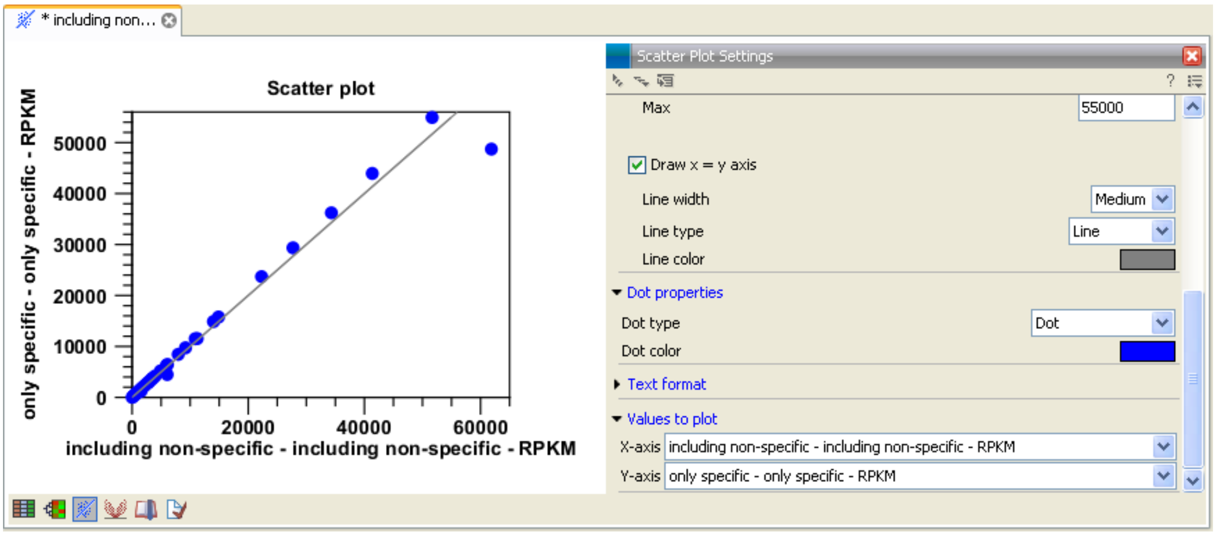
We refer to [Mortazavi et al., 2008] for an in-depth discussion of this topic.

2.11.3 The RPKM expression measure

Normalizing for sample size

The observations made from figure 2.58 lead to another important consideration when dealing with RNA-Seq analysis: you have to decide which expression measure you want to use. When you have several samples (as in this study with six different samples), these will have different numbers and qualities of reads. You will often see that there is quite a big difference between the samples in the number of reads that can be matched. This means that it can be hard to compare the expression of the same gene in different samples simply by looking at the number of reads matched (i.e. total exon reads). When comparing the groups *including non-specific* versus *only specific total*, you can see this effect too, since they have 213,569 and 199,189 mapped reads, respectively. This means that you have an asymmetry in the scatter plot when using total exon reads as the expression measure (see we could see in figure 2.58).

There is another expression measure, RPKM (Reads Per Kilobase of exon model per Million mapped reads), which seeks to normalize for the difference in number of mapped reads between samples. We will now investigate RPKM in greater detail. Go back to the scatter plot in figure 2.58. Change the values to be plotted from total exon reads to RPKM for the two samples. You should now see a scatter plot as shown in figure 2.64.



Max 55000

☑ Draw x = y axis

Line width Medium

Line type Line

Line color

Dot properties

Dot type Dot

Dot color

Text format

Values to plot

X-axis including non-specific - including non-specific - RPKM

Y-axis only specific - only specific - RPKM

Figure 2.64: *The effect of including non-specific reads compared using RPKM.*

Where figure 2.58 showed either dots falling on the $x = y$ axis or below, you now see dots falling primarily slightly above $x = y$ axis or below. This is because the RPKM takes into account that the total number of mapped reads is higher in the *including non-specific* sample than in the *only specific* sample. RPKM is defined as $RPKM = \frac{\textit{total exon reads}}{\textit{mapped reads(millions)} \times \textit{exon length (KB)}}$.

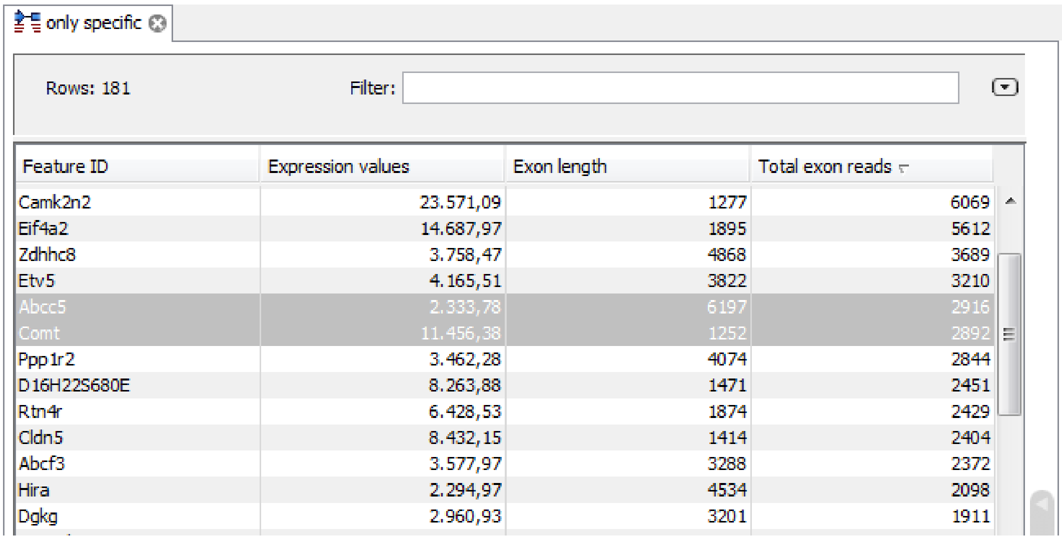
Let's investigate two of the genes in the scatter plot. First, identify the two genes at the top of the scatter plot — one above the $x = y$ axis, one below. One of them is the *Sept5* gene that we have previously investigated. This still shows higher expression in the *including non-specific sample* because of the high number of non-specific matches. The other gene is *Sst*. Switch back to the experiment table (▦) and compare the total exon reads for both samples (you can deselect sample columns under **Sample level** in the **Side Panel**, that will ease the overview). The value is 6600 and 6514, respectively, so the number of total exon reads is almost identical. What is then the reason that the PRKM value is higher for the *only specific* sample? This is because this sample has a lower number of mapped reads, and the RPKM will thus be higher (see definition of RPKM above).

Normalizing for transcript length

In a sample, if two transcripts are present in the same number of copies, and the sequencing is unbiased, you would expect the same number of reads from each transcript. But if one transcript is short and the other is long, you would expect the long transcript to yield more reads. So if you wish to compare the expression of transcripts within the same sample, you need to take the transcript length into account.

If you look at the definition of RPKM above, you can see that besides number of mapped reads, the *exon length* is also considered. The idea behind this is to make it possible to compare expression levels of different transcripts.

Open the *only specific* sample and sort the table on total exon reads, you can see that the genes *Abcc5* and *Comt* (number 15 and 16 from the top) have almost the same number of reads (2,916 and 2,892). However, their expression value measured in RPKM is 2,333.78 and 11,456.38, respectively (see figure 2.65).



Rows: 181 Filter:

Feature ID	Expression values	Exon length	Total exon reads
Camk2n2	23.571,09	1277	6069
Eif4a2	14.687,97	1895	5612
Zdhhc8	3.758,47	4868	3689
Etv5	4.165,51	3822	3210
Abcc5	2.333,78	6197	2916
Comt	11.456,38	1252	2892
Ppp1r2	3.462,28	4074	2844
D16H22S680E	8.263,88	1471	2451
Rtn4r	6.428,53	1874	2429
Cldn5	8.432,15	1414	2404
Abcf3	3.577,97	3288	2372
Hira	2.294,97	4534	2098
Dgkg	2.960,93	3201	1911

Figure 2.65: *Nearly the same number of total exon reads for two genes leads to widely different RPKM values because of the difference in transcript lengths.*

This is simply due to the difference in transcript length which you can also see in the table under **Exon length** (which sums the lengths of all the annotated exons).

Close all open views, save the experiment, and you are ready for part III.

2.12 RNA-Seq analysis part III: Exon discovery

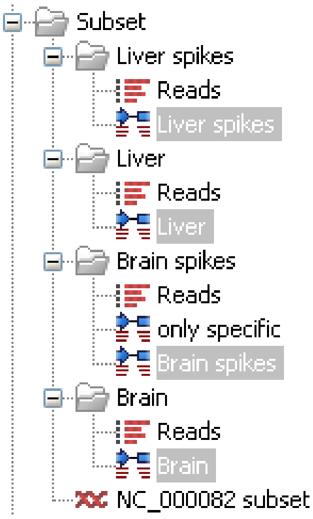
This tutorial is the third part of a series of tutorials about RNA-Seq analysis. We continue working with the data set introduced in the first tutorial.

In this tutorial we will focus on discovery of new putative exons.

2.12.1 Creating two samples for comparison

First, we will use two samples to find new putative exons expressed in one sample but not the other. In the previous tutorial, you have already created a sample called *including non-specific*. Rename this to *Brain spikes*. Then run three new RNA-Seq analyses with identical parameters but based on the *Brain*, *Liver spike* and *Liver reads*. Remember to change the **Maximum number of hits for a read** value back to 10.

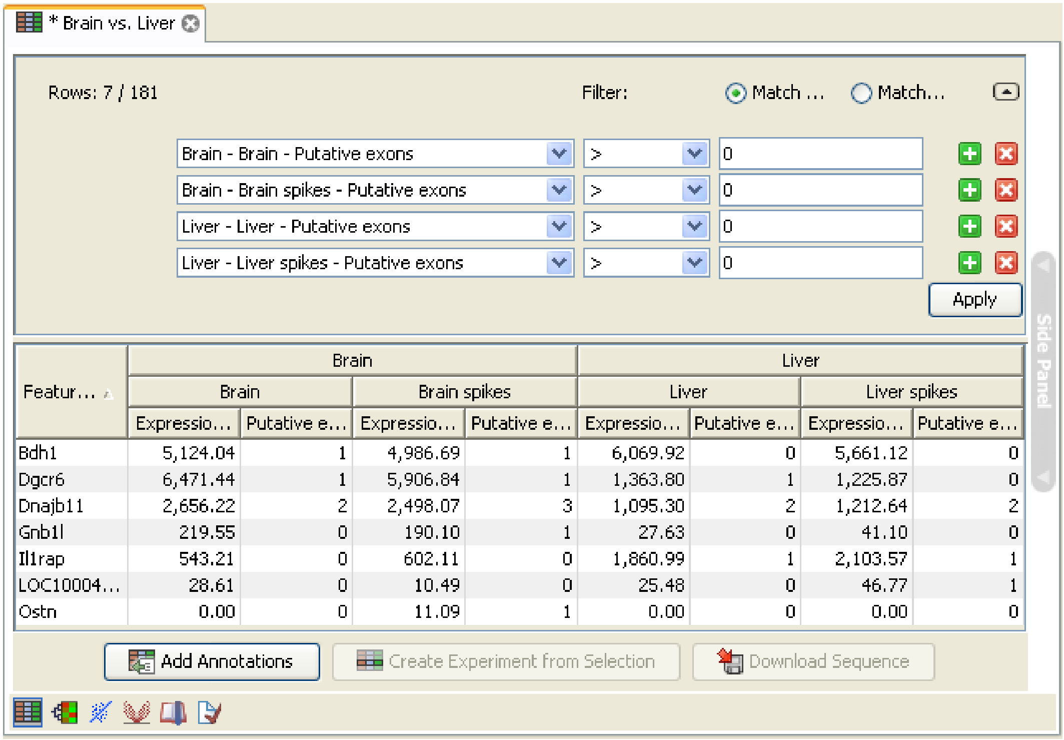
Save the results in the appropriate folders, and you should now have a folder structure like the one shown in figure 2.66.



- Subset
  - Liver spikes
    - Reads
    - Liver spikes
  - Liver
    - Reads
    - Liver
  - Brain spikes
    - Reads
    - only specific
    - Brain spikes
  - Brain
    - Reads
    - Brain
  - NC_000082 subset

Figure 2.66: *Four samples have now been created.*

Now, set up an experiment with two groups, *Brain* and *Liver* and assign the four samples to the appropriate groups. Next, adjust the settings in the **Side Panel** and the **Advanced filter** (▾) to look like figure 2.67.



Rows: 7 / 181

Filter: ● Match ... ○ Match...

Brain - Brain - Putative exons	>	0
Brain - Brain spikes - Putative exons	>	0
Liver - Liver - Putative exons	>	0
Liver - Liver spikes - Putative exons	>	0

Featur...	Brain				Liver			
	Brain		Brain spikes		Liver		Liver spikes	
	Expressio...	Putative e...	Expressio...	Putative e...	Expressio...	Putative e...	Expressio...	Putative e...
Bdh1	5,124.04	1	4,986.69	1	6,069.92	0	5,661.12	0
Dgcr6	6,471.44	1	5,906.84	1	1,363.80	1	1,225.87	0
Dnajb11	2,656.22	2	2,498.07	3	1,095.30	2	1,212.64	2
Gnb1l	219.55	0	190.10	1	27.63	0	41.10	0
Il1rap	543.21	0	602.11	0	1,860.99	1	2,103.57	1
LOC10004...	28.61	0	10.49	0	25.48	0	46.77	1
Ostn	0.00	0	11.09	1	0.00	0	0.00	0

Figure 2.67: *Comparing putative exons between the groups.*

To specify the **Advanced filter**, click the button (▾) at the top right corner of the view and click the **Add new filter criterion** (+) button to add more criteria to the filter. Notice that we want all genes that have putative exons in at least one sample so we choose the **Match any** option for the filter.

This view enables you to identify the patterns of putative exons across all the samples. You can see that there is not complete consistency between the replicates in each tissue - the brain spikes sample has in general more putative exons than the other brain sample. This may be due to a difference in the number of mapped reads for the two samples (213,569 vs 117,429). When there are less assembled reads, there will typically be less evidence for a putative exon, since you specify a minimum number of reads for a putative exon when you run the RNA-Seq analysis (default is 10).

2.12.2 Identifying new and differentially expressed splice isoforms

If you take the gene at the top of the experiment, *Bdh1*, you can see that there is a putative exon in both of the brain samples but none in the liver samples. Now open the *Brain spikes* RNA-Seq result (![icon]) and open the *Bdh1* mapping. Zoom out and you will be able to see the putative exon as shown in figure 2.68.

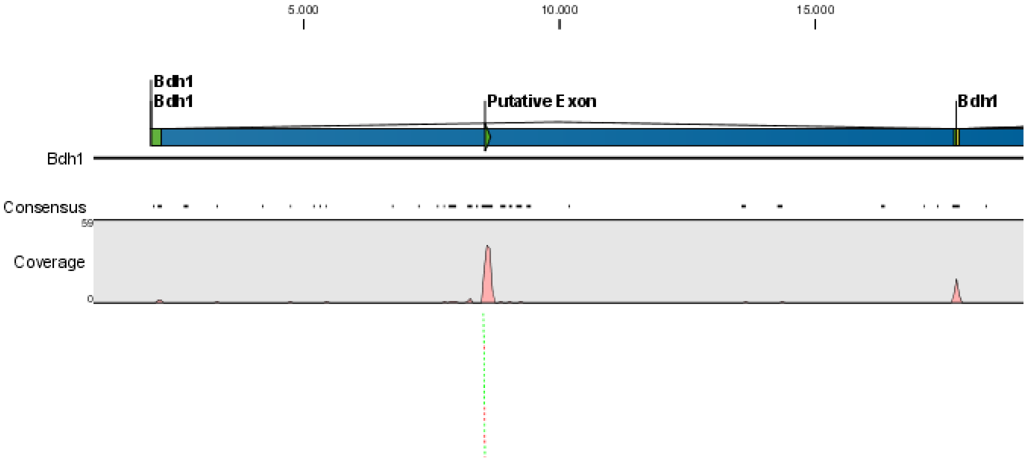


Figure 2.68: *A putative exon in the brain sample.*

If you zoom more in, you can see a very clear signal with many reads mapping in this region. The coverage level is almost identical between this and the other exons in the gene, except the first one which has very low coverage. This could indicate an alternative start site of this transcript and it seems like this new isoform is prevalent in the brain tissue.

Open the liver spikes sample and look at the *Bdh1* mapping (see figure 2.69).

Notice that the peak seen in the putative exon region of the brain sample is completely absent in the liver sample (use the position on the reference sequence to guide you - the peak should have been around position 8500. You can also use the **Search** field in the **Side Panel** to find this position). In both samples there are reads spanning the junction between the first and second exons. When you look at the first exon, you can see that the liver sample has good coverage here which indicates that the splice isoform in liver is actually the one annotated, whereas, in addition to the annotated isoform, a new splice isoform may be expressed in the brain sample.

Note that the reads within putative exons do not count in the total exon reads number (and as a consequence not in RPKM either). Also, reads are only mapped to exon-exon junctions of annotated mRNA transcripts. If you want to include these reads in the analysis, you need to annotate the original reference sequence with this new transcript by adding a new mRNA annotation which is identical to the existing one but with the new exon included. We cannot

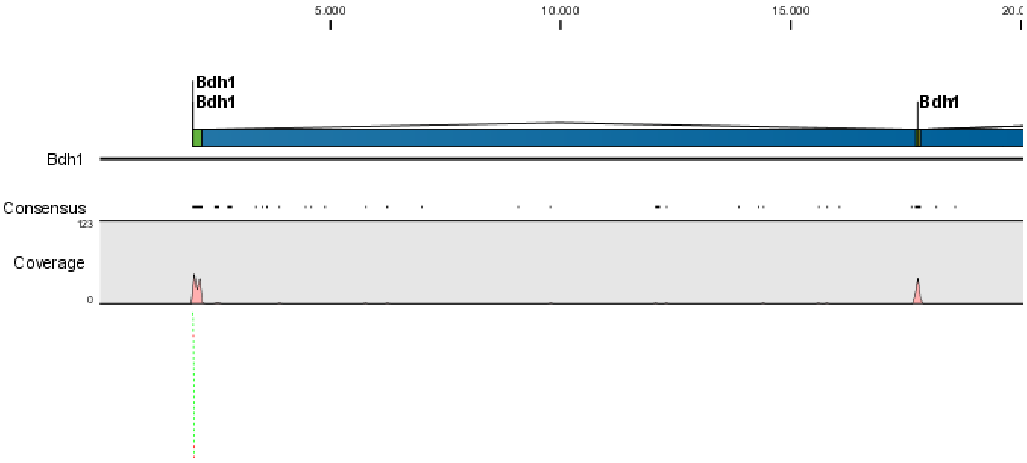


Figure 2.69: *In the liver sample, there is no putative exon.*

be sure that this is right, but judging from the coverage levels of the other exons, it would be a plausible explanation. Then run the RNA-Seq analysis again using the modified reference sequence.

Close all open views, and you are ready for part IV.

2.13 RNA-Seq analysis part IV: Spikes and quality control

This tutorial is the fourth part of a series of tutorials about RNA-Seq analysis. We continue working with the data set introduced in the first tutorial, although in this tutorial we are no longer considering only a subset of the data.

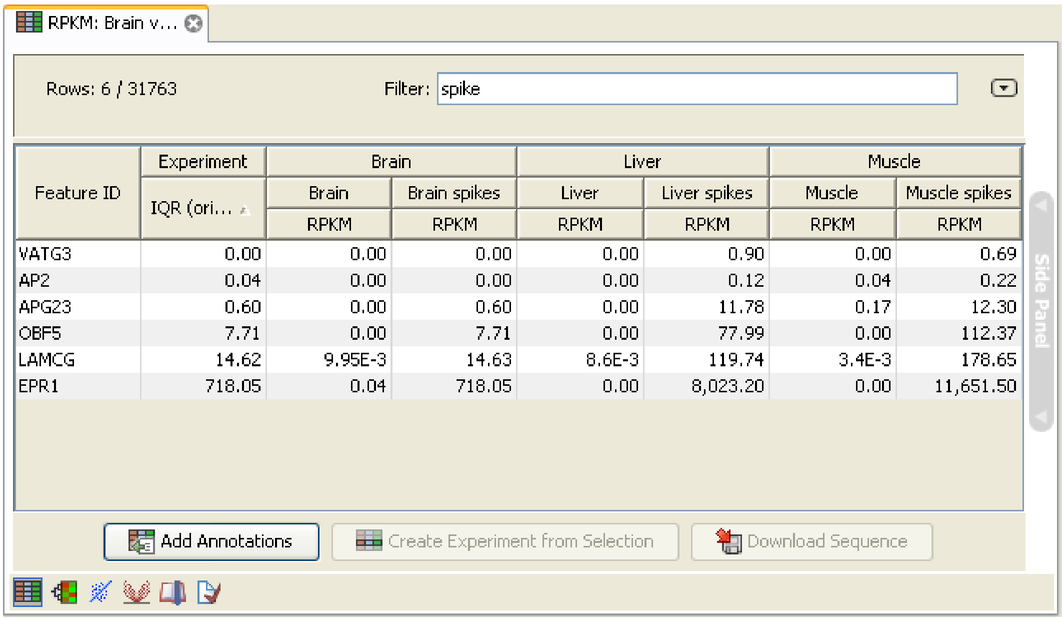
In this tutorial we will focus on quality control. First, we will examine spike-ins in the data set, and second you will learn about general quality control tools.

2.13.1 Inspecting the spike reads

The data set includes six samples that come from three groups corresponding to the three different types of tissue. Within each group of samples, one sample has six spike-in genes, where controlled amounts of mRNA from *Arabidopsis* and phage lambda templates have been introduced in the sample prior to sequencing. We will now inspect the six spike-in genes and check if they are expressed in the spiked samples only as we would expect.

For this and the rest of the tutorial, we will work on the full data set. The data set is so large that on a powerful eight-core workstation computer with 32 GB RAM, it takes more than an hour to run one sample. Therefore, we have performed the analysis beforehand and included the final result with the files you downloaded in the first tutorial. The full RNA-Seq results with mappings of all the mouse genes take up too much space, so they are not included. Instead, we have set up experiments with all six samples.

Open the experiment in the *RPKM* folder and type "spike" in the filter to only include the six spike genes (when we ran the full analysis, we made an artificial sequence called "Spike chromosome" containing the six spike genes). Next, click the **IQR** column header to sort the genes according to the interquartile range of expression. You should now have a view similar to figure 2.70.



Rows: 6 / 31763 Filter: spike

Feature ID	Experiment	Brain		Liver		Muscle	
	IQR (ori...	Brain	Brain spikes	Liver	Liver spikes	Muscle	Muscle spikes
		RPKM	RPKM	RPKM	RPKM	RPKM	RPKM
VATG3	0.00	0.00	0.00	0.00	0.90	0.00	0.69
AP2	0.04	0.00	0.00	0.00	0.12	0.04	0.22
APG23	0.60	0.00	0.60	0.00	11.78	0.17	12.30
OBF5	7.71	0.00	7.71	0.00	77.99	0.00	112.37
LAMCG	14.62	9.95E-3	14.63	8.6E-3	119.74	3.4E-3	178.65
EPR1	718.05	0.04	718.05	0.00	8,023.20	0.00	11,651.50

Figure 2.70: *Comparing the expression of the spikes.*

The important thing to check here is that we see expression of the spike genes in the spike samples but not in the rest. The six spike transcripts were added in different concentrations to the spike samples, and this is why we see the varying levels of expression.

Select all six rows, and switch to the scatter plot ( ) to visualize this trend. Choose to compare expression value of *Brain spikes* and *Brain*, and you will be able to see some red points in the scatter plot. Change the **Dot type** to **Dot** in the **Side Panel** to see the red dots more clearly. The scatter plot is shown in figure 2.71.

At first glance, you only see one clear outlier at the bottom, whereas the other red dot (or rather, 'dots') seem to have more similar expression in the two samples (close to zero in both). But if you **Zoom in** ( ) on these dots, you can see that 3 of them are outliers, although to a smaller extent (see figure 2.72).

2.13.2 Checking within and between group variability

We have now confirmed that the spike controls look fine (you can check the other two groups in the scatter plot if you like). The next step in our quality control efforts is to check whether the overall variability of the samples reflect their grouping. In other words we want the samples from the same group to be relatively homogenous and distinguishable from the samples of the other groups.

Box plot

To examine and compare the overall distribution of the expression values in the samples you may use a **Box plot** ( ). Box plots can be used to get an overall impression of the locations of the distributions, and to some extend the spread of the distributions.

First, create a box plot based on the RPKM-based experiment:

Toolbox | Expression Analysis ( ) | Quality Control | Create Box Plot ( )

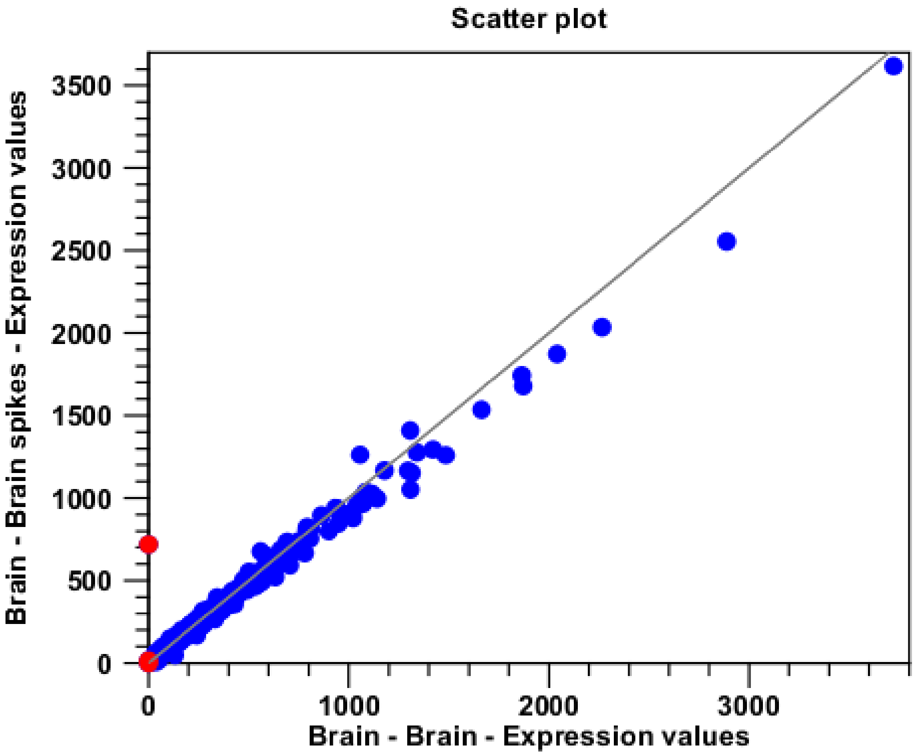


Figure 2.71: *Comparing the expression of the spikes in a scatter plot.*

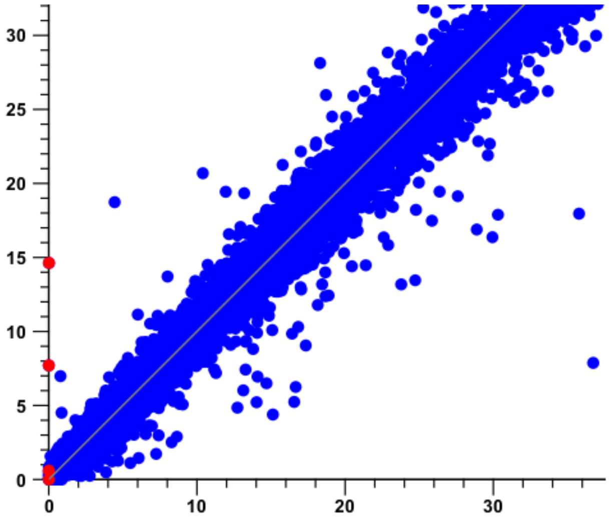


Figure 2.72: *Comparing the expression of the spikes in a scatter plot, zoomed in.*

Select the experiment that is based on RPKM expression values and click **Next** and **Finish**.

The box plot is shown in figure 2.73.

The median and mean lines (you can display the mean in the **Side Panel**) show the median and mean expression values in the samples and the boxes extend from the p'th to the 100-p'th percentile of the sets of expression values in the samples. Thus, the box plot view is rather

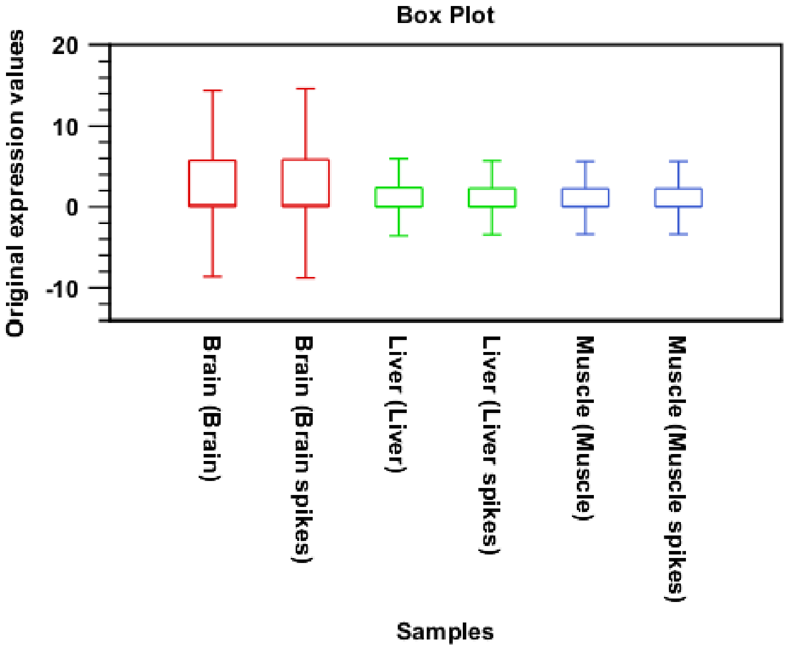


Figure 2.73: *A box plot of the 6 samples in the experiment, colored by group.*

sensitive to the choice of the percentile value, and you may get a better impression of how the distributions compare by trying different percentiles.

The distributions of the expression values are dominated by a lot of really small values and much fewer but much larger values. To diminish the effect on the box plots of the few very high values, you can square root transform the values and create box plots for the transformed values. First, transform the values in the experiment:

> **Toolbox | Expression Analysis ( ) | Transformation and Normalization | Transform ( )**

Select the experiment, click **Next** and select **Square root transformation**. Click **Finish**. Now, create a new box plot, but this time make sure to select **Transformed expression values** in the second step. Figure 2.74 shows the box plot before and after square root transformation.

After square root transformation, the distributions appear more a bit more similar.

Now, create similar box plots for the experiment based on total exon reads. You can see the result in figure 2.75 that again shows the box plot before and after square root transformation, this time based on total exon reads rather than RPKM.

Overall, the box plots indicate that the *locations* of the distributions of the expression values in the samples are similar both for RPKM and for Total Exon Reads, but there is considerable difference in the *spread* of the values. The distributions of RPKM values for samples for the same tissue are highly similar. This is expected, as the samples are technical replicates and the sample size is factored out in the RPKM (see part III of the tutorials). The high variability of the 'Total exon reads' counts is obviously related to the numbers of (mapped) reads in the samples.

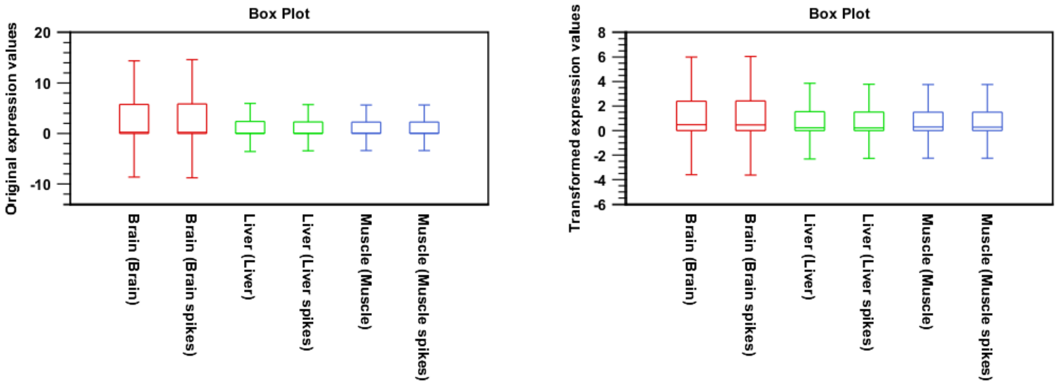


Figure 2.74: *To the left: original box plot based on RPKM values. To the right: box plot based on square root transformed expression values.*

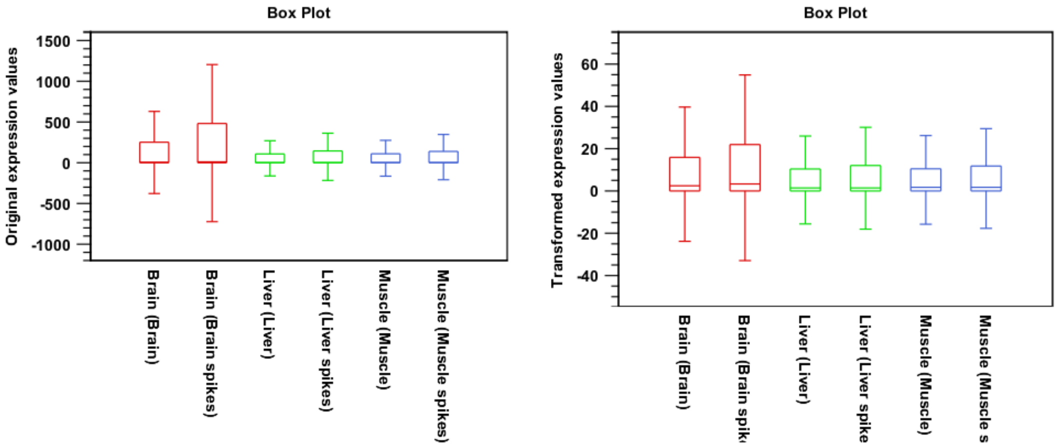


Figure 2.75: *To the left: original box plot based on total exon reads. To the right: box plot based on square root transformed expression values.*

Principal component analysis (PCA)

We continue to work with the experiment based on RPKM expression values. Next, we perform a **Principal Component Analysis (PCA)**:

> **Toolbox** | **Expression Analysis (**▦**)** | **Quality Control** | **Principal Component Analysis (**▦**)**

Select the experiment, click **Next** and **Finish**. This will create a PCA plot as shown in figure 2.76.

The plot shows the projection of the samples onto the two-dimensional space spanned by the first and second principal component. (These are the orthogonal directions in which the data exhibits the largest and second-largest variability).

The dots are colored according to the groups, and they group very nicely in the plot (the two red red dots are on top of each other).

You can display the names of the samples in the plot using the settings under **Dot properties** in the **Side Panel** to the right of the view (see the result in figure 2.77).

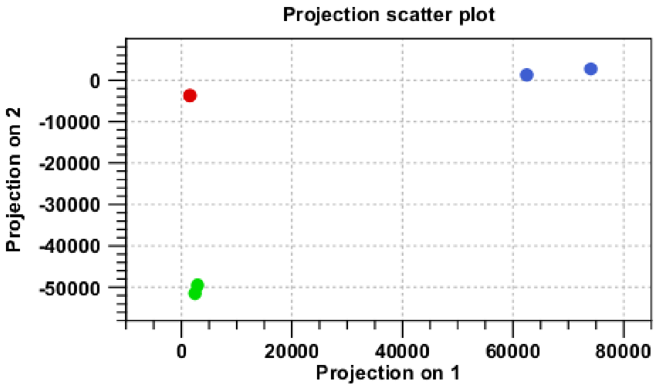


Figure 2.76: *A principal component analysis colored by group.*

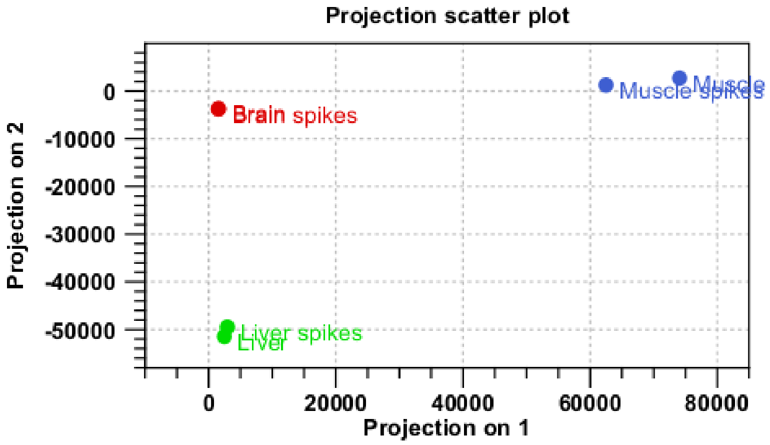


Figure 2.77: *Displaying the sample names.*

This PCA was done on RPKM-based expression values. As you saw in the previous tutorial, the RPKM normalizes for the sample size. In order to show the importance of doing this for this kind of analysis, perform a new PCA on the experiment located in the *total exon reads* folder.

The result is shown in figure 2.78.

Although the replicates still group together, the grouping is not near as clear as in figure 2.77 which uses the RPKM-based expression values.

Hierarchical clustering

In order to complement the principal component analysis, we will also do a hierarchical clustering of the samples to see if the samples cluster in the groups we expect:

> **Toolbox** | **Expression Analysis ()** | **Quality Control** | **Hierarchical Clustering of Samples ()**

Select the experiment from the *RPKM* folder and click **Next**. Leave the parameters at their default and click **Finish**.

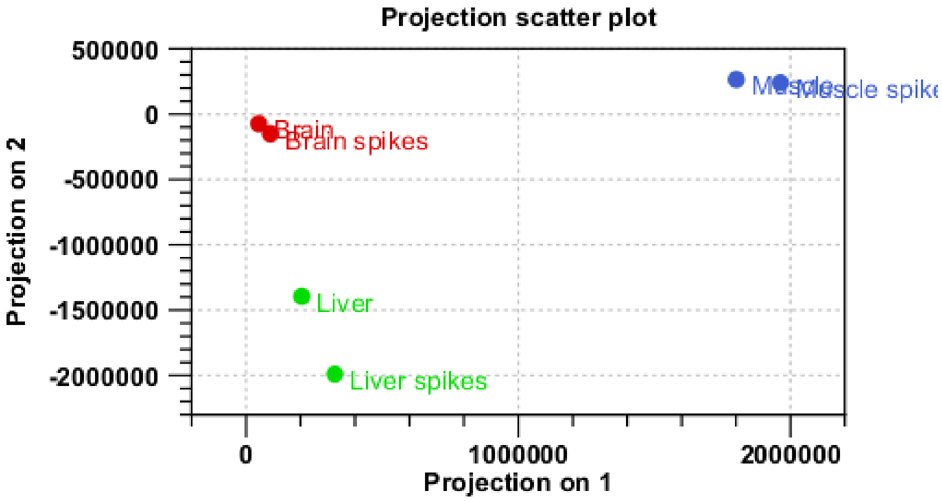


Figure 2.78: *PCA plot using total exon reads.*

This will display a heat map showing the clustering of samples at the bottom (see figure 2.79).

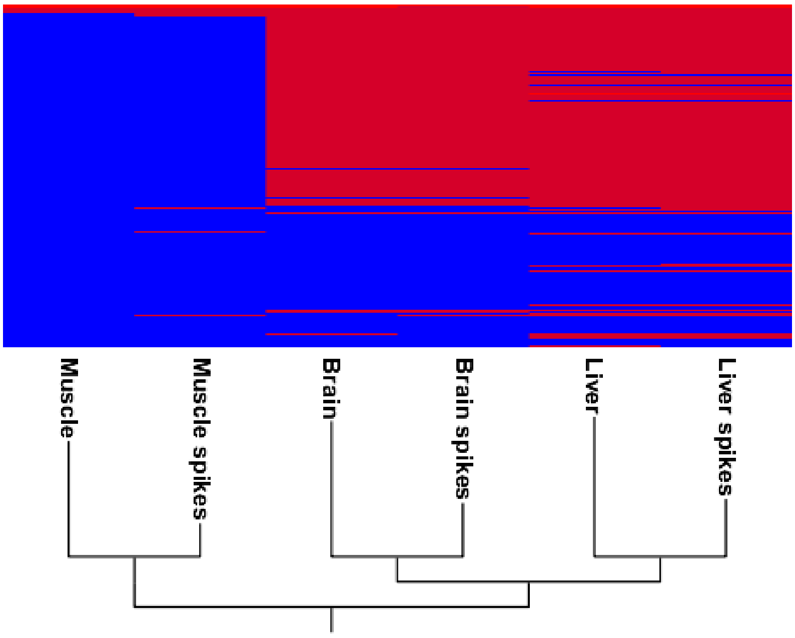


Figure 2.79: *Sample clustering. Adjusting the settings in the Side Panel to put the sample names at the bottom.*

The replicates cluster nicely together as expected, and it looks like the pattern of expression is more similar between brain and liver than between muscle and the other two groups.

Note that the heat map is not a new element to be stored in the **Navigation Area** - it is just another way of looking at the experiment (note the buttons to switch between different views in figure 2.80.

Figure 2.80: *Different views on an experiment.*

As a conclusion to this fourth tutorial on RNA-Seq, we can confirm that the spike signals are clear and unambiguous, and we can conclude that both the PCA and hierarchical clustering shows that the replicates are fairly homogenous. Just as in the previous tutorial, we can see the effect of using RPKM over total exon reads as expression measure. When doing this kind of quality control, it may be advisory to use the RPKM expression value as this is implicitly standardized between samples (see tutorial part II). Alternatively, you can choose to use the total exon reads counts and then normalize them before performing quality control (you find the **Normalize** () tool here: Toolbox -> Expression Analysis -> Transformation and Normalization).

One of the reasons why the samples cluster so nicely is that there is no biological variation in the samples. The two samples in each group are technical replicates, so what we are really measuring is the quality of the method. In experiments with biological replicates, you would expect to see much more intra-group variability.

2.14 Tutorial: Small RNA analysis using Illumina data

This tutorial shows how to go through the initial parts of analyzing a small RNA data set. It shows how to analyze one sample including how to trim off adapter sequences and extract the small RNAs, how to count the small RNAs, inspect and check the results and finally how to annotate the small RNAs to identify known miRNAs and other non-coding RNAs.

The tutorial is based on the study published in [Stark et al., 2010]. In this study, 12 samples from melanoma cell cultures were sequenced using an Illumina Genome Analyzer II. In this tutorial, we will analyze one of these samples, the MELB sample, which represents a primary melanocyte cell.

2.14.1 Downloading and importing the raw data

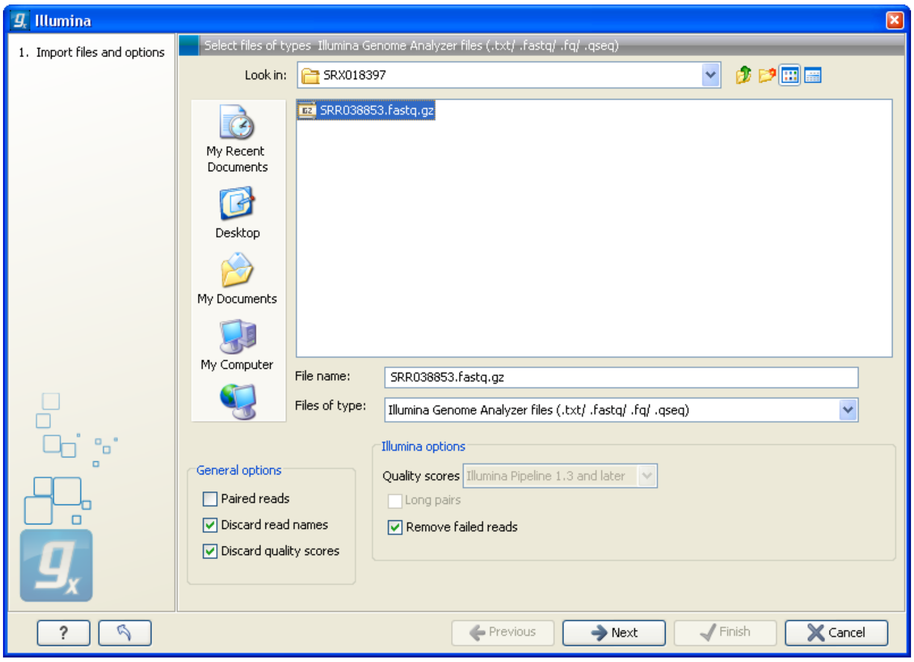
First, download the fastq file containing all the raw data from this sample: `http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=dload&run_list=SRR038853&format=fastq`.

Now, import the data set using the Illumina importer:

File | **Import High-throughput Sequencing** | **Illumina**

This will bring up the dialog shown in figure 2.81.

Select the `SRR038853.fastq.gz`. Make sure the **Discard quality scores** and **Discard read names** checkboxes are checked. Information about quality scores and read names are not used in this analysis anyway, so it will just take up disk space when importing the data. Click **Next**, choose to **Save** and click **Finish**.



Illumina
1. Import files and options
Select files of types Illumina Genome Analyzer files (.txt/ .fastq/ .fq/ .qseq)
My Recent Documents
Desktop
My Documents
My Computer
SRR038853.fastq.gz
?
Previous
Next
Finish
Cancel

Look in: SRX018397

File name: SRR038853.fastq.gz

Files of type: Illumina Genome Analyzer files (.txt/ .fastq/ .fq/ .qseq)

General options

□ Paired reads

☑ Discard read names

☑ Discard quality scores

Illumina options

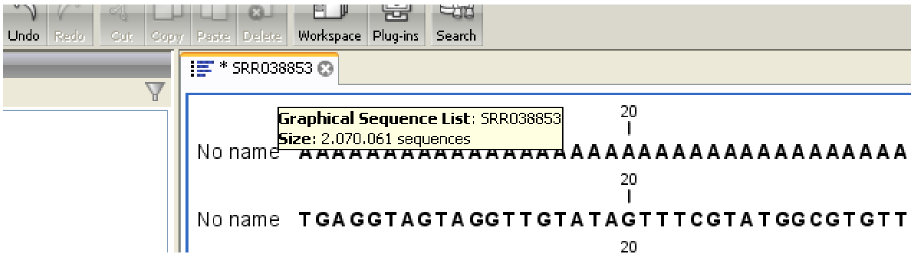
Quality scores Illumina Pipeline 1.3 and later

□ Long pairs

☑ Remove failed reads

Figure 2.81: *Selecting the fastq file for import.*

After a short while, the reads have been imported. Open the file you imported by double-clicking and place your mouse on the tab. After one second, you will see a small tool tip with information about the number of reads in the file as shown in figure 2.82.



Graphical Sequence List: SRR038853
Size: 2.070.061 sequences

No name AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

No name TGAGGTAGTAGGTTGTATAGTTTCGTATGGCGTGTT

Figure 2.82: *This data set contains about two million reads.*

Close the view again.

2.14.2 Trimming adapters and counting the reads

The next step in the analysis is to trim off the partial adapter sequences and subsequently to count how many copies there are of each of the resulting small RNAs.

> **Toolbox | High-throughput Sequencing ( ) | Small RNA Analysis ( ) |Extract and Count ( )**

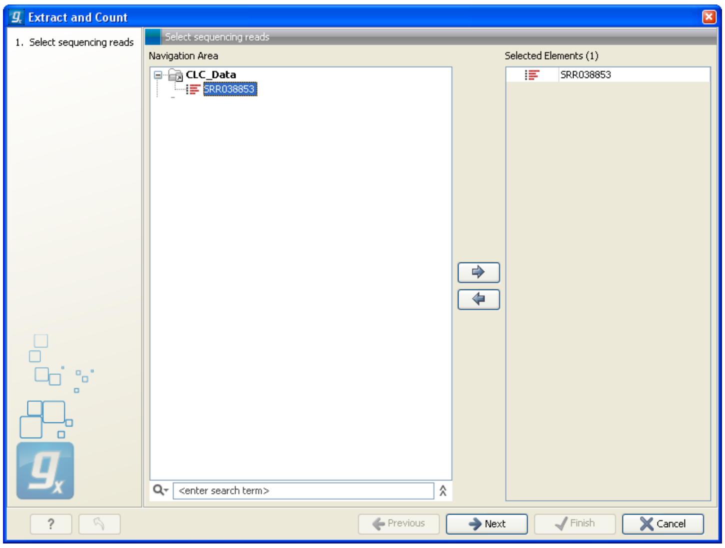
This opens a dialog where you select the SRR038853 sample as shown in figure 2.83.

Click **Next** when the data is listed in the right-hand side of the dialog.

You are now presented with the dialog shown in figure 2.84.

Make sure the checkbox is selected and click **Next**.

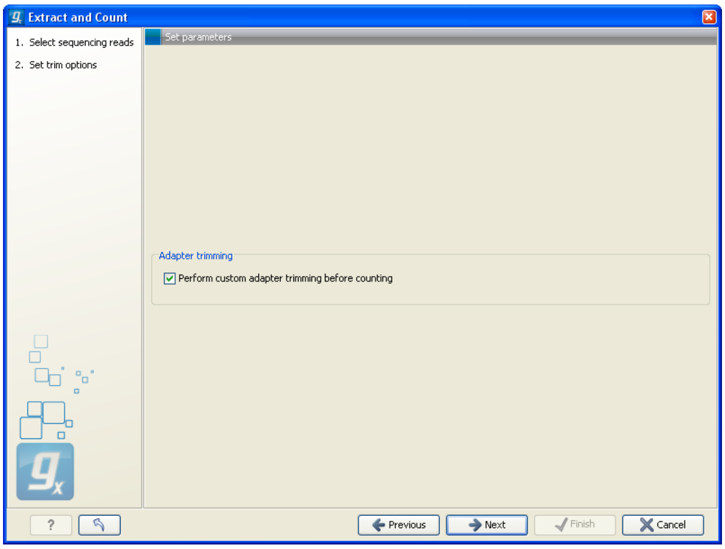
You will now see the dialog shown in figure 2.85.



CLC_Data
SRR038853

Selected Elements (1): SRR038853

Figure 2.83: *Selecting the sample for extracting and counting the small RNAs.*



Adapter trimming

☑ Perform custom adapter trimming before counting

Figure 2.84: *Choosing to trim for adapter sequence.*

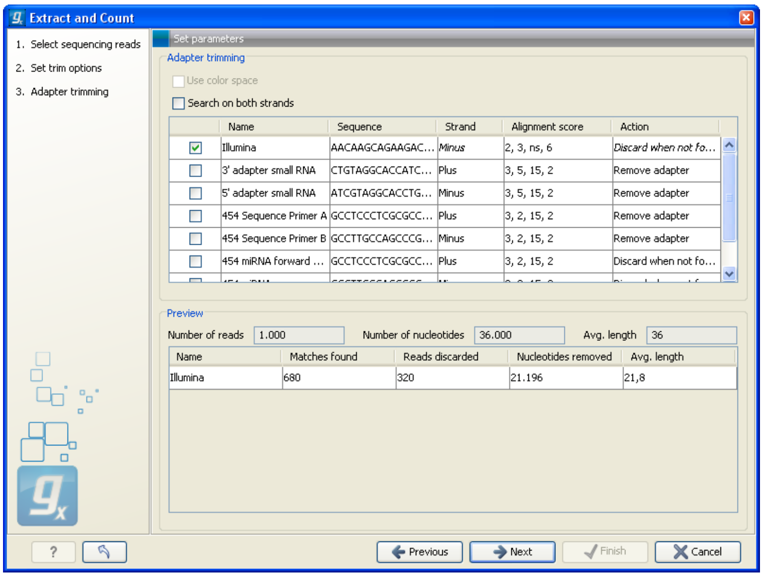
In the list of adapter sequences, select the Illumina adapter. You can see in the preview panel below how many matches that are found for this adapter among the first 1000 reads in the input file. We will see more statistics on this for the full data set later on - this preview is just intended to support the user when defining the adapter trim setting.

Since the trim settings are already set right, click **Next**. Note that you could have changed both Strand, Alignment score and action in this panel by clicking/double-clicking the adapter.

You will now see the dialog shown in figure 2.86.

The most important choices in this dialog are that you can set a minimum and maximum length on the tags that you want to include when counting, and that you can decide how many copies there have to be in order to include the tag in the output. Leave these settings at the default and click **Next**.

This will allow you to specify the output options as shown in figure 2.87.



1. Select sequencing reads
2. Set trim options
3. Adapter trimming

Adapter trimming

□ Use color space

□ Search on both strands

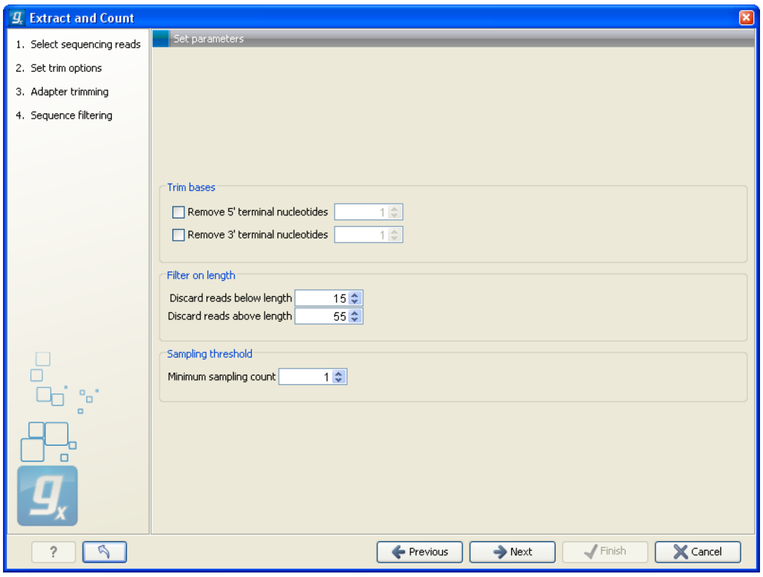
	Name	Sequence	Strand	Alignment score	Action
☑	Illumina	AACAAGCAGAAGAC...	*Minus*	2, 3, ns, 6	*Discard when not fo...*
□	3' adapter small RNA	CTGTAGGCACCATC...	Plus	3, 5, 15, 2	Remove adapter
□	5' adapter small RNA	ATCGTAGGCACCTG...	Minus	3, 5, 15, 2	Remove adapter
□	454 Sequence Primer A	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Remove adapter
□	454 Sequence Primer B	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Remove adapter
□	454 miRNA forward ...	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Discard when not fo...

Preview

Number of reads 1.000 Number of nucleotides 36.000 Avg. length 36

Name	Matches found	Reads discarded	Nucleotides removed	Avg. length
Illumina	680	320	21.196	21,8

Figure 2.85: *Trimming the raw data for adapter sequence.*



1. Select sequencing reads
2. Set trim options
3. Adapter trimming
4. Sequence filtering

Trim bases

□ Remove 5' terminal nucleotides 1

□ Remove 3' terminal nucleotides 1

Filter on length

Discard reads below length 15

Discard reads above length 55

Sampling threshold

Minimum sampling count 1

Figure 2.86: *Adjusting options for counting the small RNAs.*

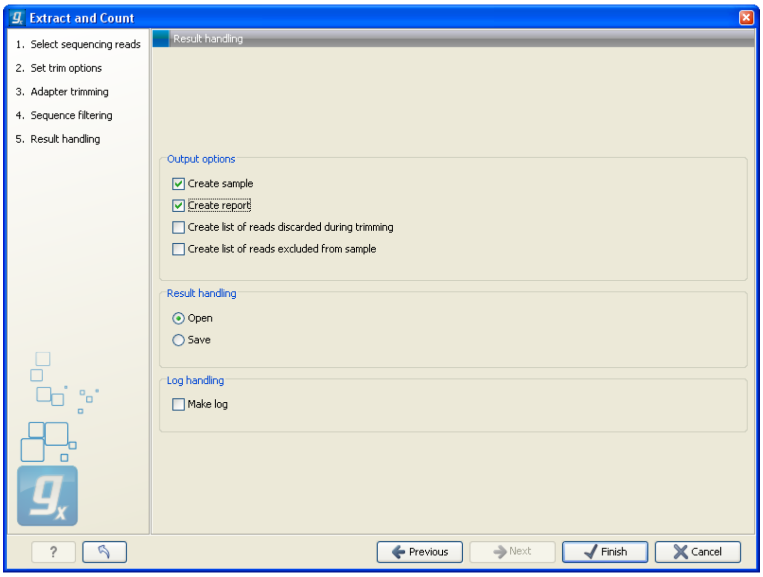
The default is to output a **Sample** which is the table of all the small RNAs and their counts, and to create a report showing summary statistics. Leave the settings at default and click **Finish**

2.14.3 Interpreting the adapter trim report

Once the analysis is complete, two tabs will be opened. First, we take a look at the report.

The top part of the report is shown in figure 2.88.

The report is meant to be used as a quality check, mainly to see that the adapter trimming worked as expected. In this example, it shows that out of 2 million reads, 1.7 million reads passed the adapter trim. The trim settings meant that if no adapter sequence was found, the read would have been discarded. So this means that (part of) the adapter sequence was found in all these 1.7 million reads.



Output options

☑ Create sample

☑ Create report

□ Create list of reads discarded during trimming

□ Create list of reads excluded from sample

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 2.87: *Selecting the results to output.*

There is also a graph showing a distribution of the read lengths after trimming. In this example, there is a very nice distribution with a peak around 22 bp which is expected for miRNAs.

2.14.4 Investigating the small RNA sample

Save and close the report, and you should now see the small RNA sample as shown in figure 2.89.

There are 88,460 unique small RNAs in the sample. You can filter and sort the sample, and you can extract subsets using the buttons at the bottom of the view. As an example, we will try to open the trimmed reads of one of the small RNAs: Sort the table on Length (clicking the column header) and click the row at the top. Then click the **Extract Reads** button and click **Finish** in the dialog that is opened. You should now be able to see the original read sequence with a trim annotation as shown in figure 2.90.

Clicking the **Double stranded** checkbox in the Side Panel to the right, you can see the minus strand as well, and you can see that the adapter sequence, `CAAGCAGAAGACGGCATACGA` has a perfect match here.

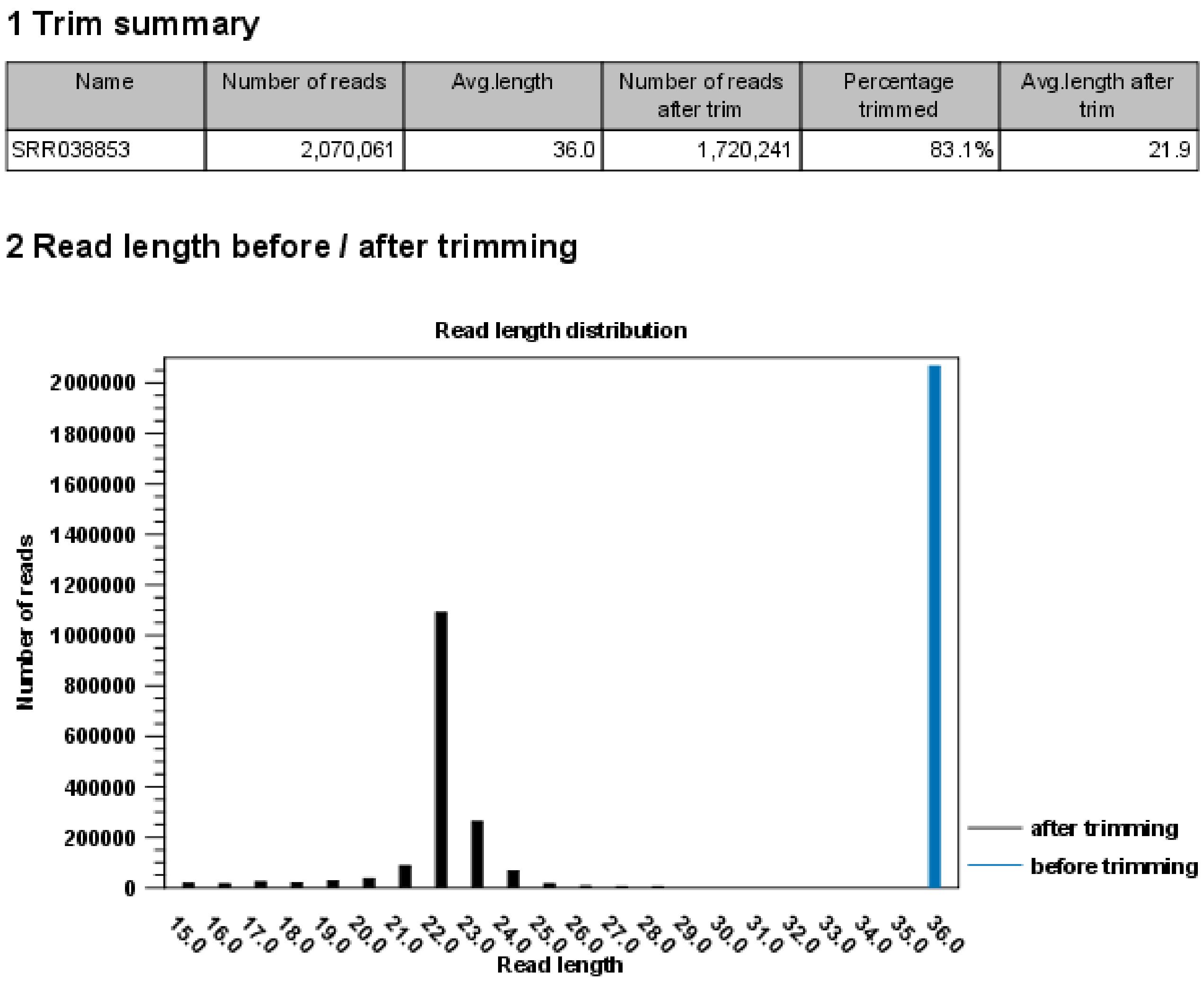
2.14.5 Downloading miRBase and annotating the sample

The next step in the analysis is to annotate the small RNA sample to identify known small RNAs. We use two sources for the annotation: first, miRBase is used to identify known miRNAs and second a set of other known non-coding RNAs.

The Workbench lets you download the latest version of miRBase directly:

> **Toolbox** | **High-throughput Sequencing** ( ) | **Small RNA Analysis** ( ) |**Download miRBase** ( )

Choose **Save**, **Next** and **Finish**. Next, download and **Import** the set of other non-coding RNAs from `ftp://ftp.ensembl.org/pub/release-57/fasta/homo_sapiens/ncrna/Homo_sapiens.GRCh37.57.ncrna.fa.gz`. You should now have the two annotation files



1 Trim summary

Name	Number of reads	Avg.length	Number of reads after trim	Percentage trimmed	Avg.length after trim
SRR038853	2,070,061	36.0	1,720,241	83.1%	21.9

2 Read length before / after trimming

Figure 2.88: *The small RNA counting report.*

represented as sequence lists in the Workbench, and you are ready to start the annotation:

> **Toolbox** | **High-throughput Sequencing ( )** | **Small RNA Analysis ( )** |**Annotate and Merge Counts ( )**

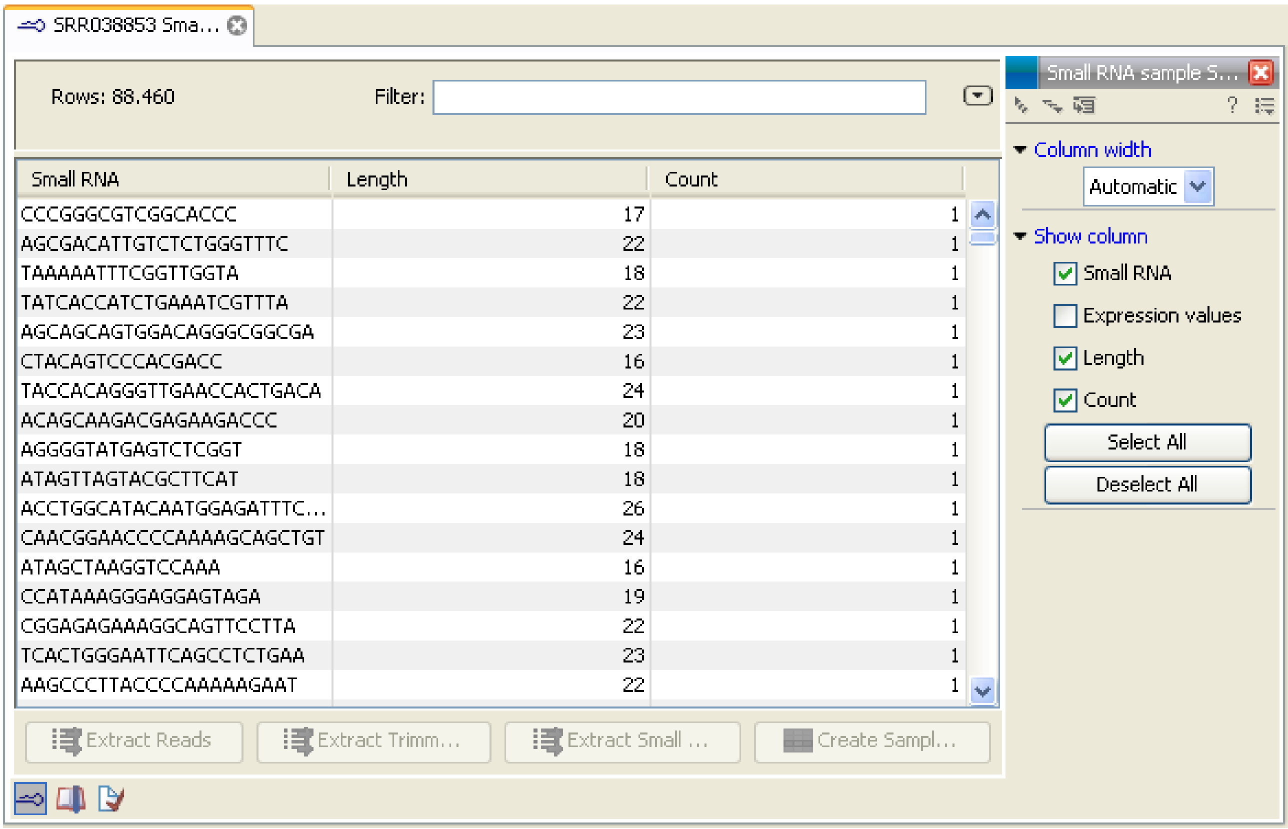
This opens a dialog where you select the `SRR038853 Small RNA sample` as shown in figure 2.91.

Click **Next** when the data is listed in the right-hand side of the dialog.

You are now presented with the dialog shown in figure 2.92.

At the top, select to use miRBase and select ( ) the miRBase file that you downloaded previously. Below, check the **Use other resource** and select ( ) the `Homo_sapiens.GRCh37.57.ncrna` file that you imported previously.

The miRBase file contains a list of precursor sequences with specification of the mature and in some cases the mature * regions. This information is used to categorize the annotated small RNAs. The **Other resources** does not include this kind of information and is used here in order



Rows: 88.460 Filter:

Small RNA	Length	Count
CCCGGGCGTCGGCACCC	17	1
AGCGACATTGTCTCTGGGTTTC	22	1
TAAAAATTTCGGTTGGTA	18	1
TATCACCATCTGAAATCGTTTA	22	1
AGCAGCAGTGGACAGGGCGGCGA	23	1
CTACAGTCCCACGACC	16	1
TACCACAGGGTTGAACCACTGACA	24	1
ACAGCAAGACGAGAAGACCC	20	1
AGGGGTATGAGTCTCGGT	18	1
ATAGTTAGTACGCTTCAT	18	1
ACCTGGCATACAATGGAGATTTC...	26	1
CAACGGAACCCCAAAAGCAGCTGT	24	1
ATAGCTAAGGTCCAAA	16	1
CCATAAAGGGAGGAGTAGA	19	1
CGGAGAGAAAGGCAGTTCCTTA	22	1
TCACTGGGAATTCAGCCTCTGAA	23	1
AAGCCCTTACCCCAAAAAGAAT	22	1

Column width: Automatic

Show column
- ☑ Small RNA
- □ Expression values
- ☑ Length
- ☑ Count

Select All

Deselect All

Figure 2.89: *The small RNA sample.*

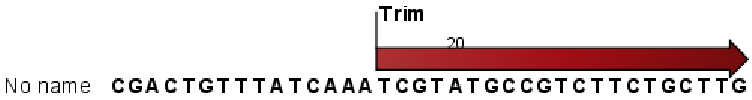


Figure 2.90: *A sequencing read displaying the trim annotation.*

to identify known small RNAs that are not miRNAs. Note that you could include several sequence lists here if you have other sources of non-coding small RNAs.

Click **Next** will show the dialog shown in figure 2.93.

Select first *Homo sapiens* and second *Mus musculus*. The sample is human, so that should be the first priority as annotation source, and mouse should be second in the list. Since there may be miRNAs that have not yet been identified in human but have an ortholog in mouse, it is interesting to include the mouse miRNAs as well.

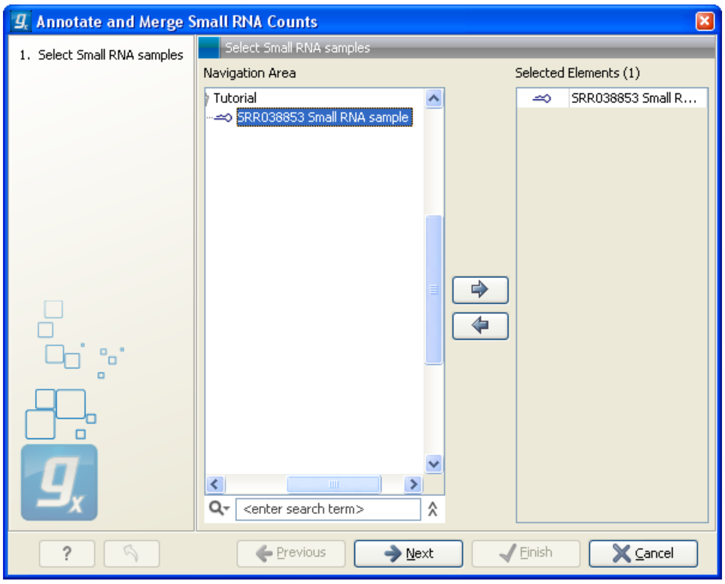
Click **Next** will show the dialog in figure 2.94.

Leave these settings at their default and click **Next** to display the dialog shown in figure 2.95.

Make sure all options except the unannotated sample are checked and click **Finish**.

2.14.6 Analyzing the annotated samples

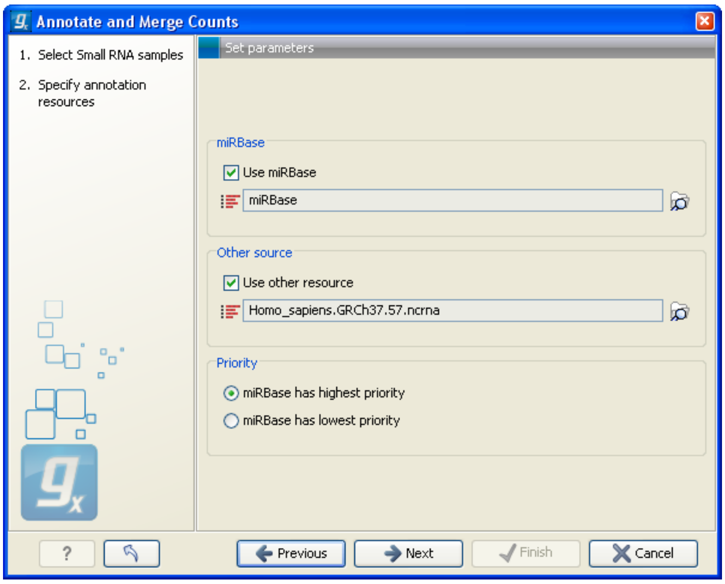
For a detailed description of the output, please have a look in the user manual (press **F1** to display). In this example we focus on a few specific miRNAs that will illustrate how the annotation



Selected Elements (1)

SRR038853 Small R...

Figure 2.91: *Selecting the sample for annotating the small RNAs.*


Annotate and Merge Counts
1. Select Small RNA samples
2. Specify annotation resources
Set parameters
?
Previous
Next
Finish
Cancel

miRBase

☑ Use miRBase

miRBase

Other source

☑ Use other resource

Homo_sapiens.GRCh37.57.ncrna

Priority

● miRBase has highest priority

○ miRBase has lowest priority

Figure 2.92: *Setting miRBase and the other non-coding RNAs as annotation source.*

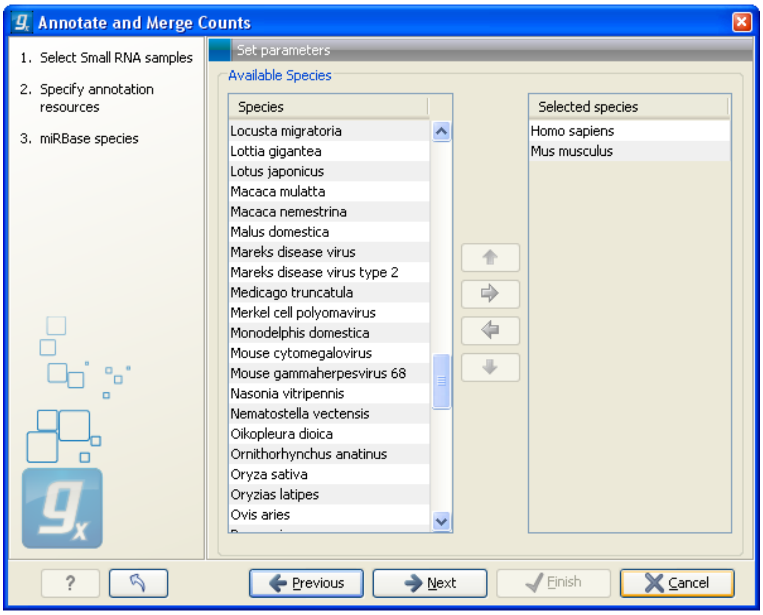
and grouping of samples work and show some of the possibilities you have for interacting with the data.

Looking at the grouped sample

Save (💾) and close all the views except for the un-grouped annotated sample (⊸). We want to look at *mir-29a*, so type this into the filter at the top of the table as shown in figure 2.96.

This will list all the tags (414 out of 31,841) that have been mapped to the mir-29a precursor sequence from miRBase. If you sort the table by **Count** (clicking the count column header), you can see that most of these are exact matches of the mature miRNA. The rest are variants and length variants.

For expression analysis, it can make sense to look at all the variants of the same miRNA as one en-



Annotate and Merge Counts

1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species

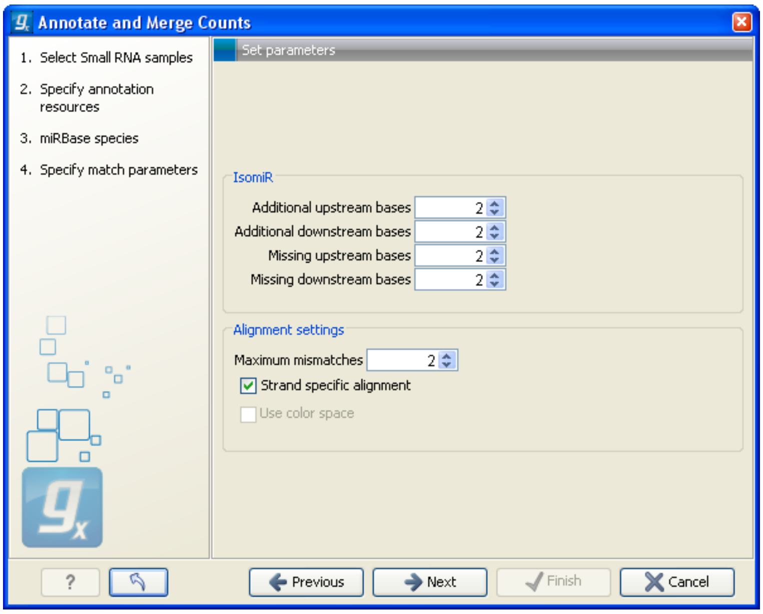
Set parameters

Available Species

Species
Locusta migratoria
Lottia gigantea
Lotus japonicus
Macaca mulatta
Macaca nemestrina
Malus domestica
Mareks disease virus
Mareks disease virus type 2
Medicago truncatula
Merkel cell polyomavirus
Monodelphis domestica
Mouse cytomegalovirus
Mouse gammaherpesvirus 68
Nasonia vitripennis
Nematostella vectensis
Oikopleura dioica
Ornithorhynchus anatinus
Oryza sativa
Oryzias latipes
Ovis aries

Selected species
Homo sapiens
Mus musculus

Figure 2.93: *Prioritizing species for annotation.*



Annotate and Merge Counts

1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species
4. Specify match parameters

Set parameters

IsomiR

Additional upstream bases: 2
Additional downstream bases: 2
Missing upstream bases: 2
Missing downstream bases: 2

Alignment settings

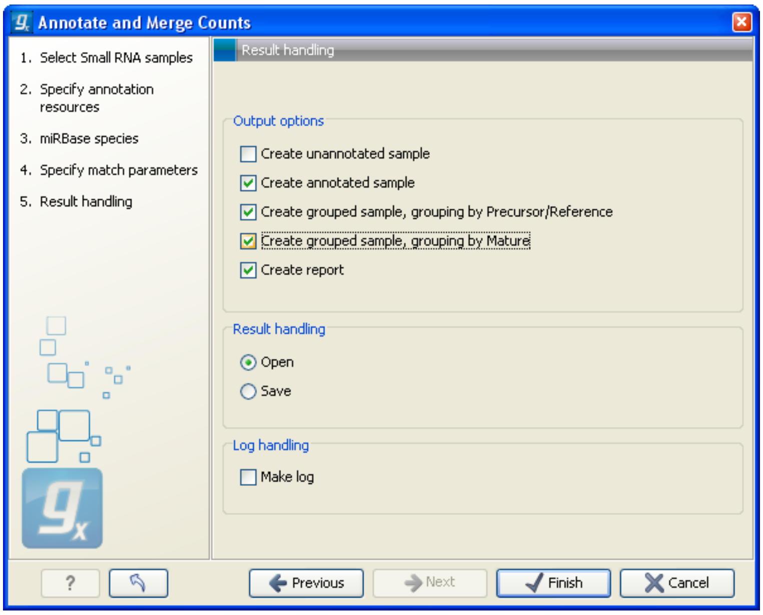
Maximum mismatches: 2
☑ Strand specific alignment
□ Use color space

Figure 2.94: *Thresholds for annotating.*

tity rather than 414 as it is the case here. Open the `SRR038853 Small RNA sample grouped` (▦) and type in mir-29a in the filter. You now have one line representing all the tags that have been annotated with mir-29a with a total count of 36,600. The number of reads in different categories are shown, e.g. 30,689 for the exact mature corresponding to the number from the ungrouped sample in figure 2.96. You also see a few tags annotated with the mouse ortholog, but this could be noise due to sequencing errors.

Double-click the human mir-29a row to open the alignment of all the tags to the precursor sequence (see figure 2.97).

The tags are colored to reflect the counts which are also shown in numbers next to the name to the left. Since the exact mature is very dominant in terms of count, it is the only one standing



Annotate and Merge Counts

1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species
4. Specify match parameters
5. Result handling

Result handling

Output options

☐ Create unannotated sample

☑ Create annotated sample

☑ Create grouped sample, grouping by Precursor/Reference

☑ Create grouped sample, grouping by Mature

☑ Create report

Result handling

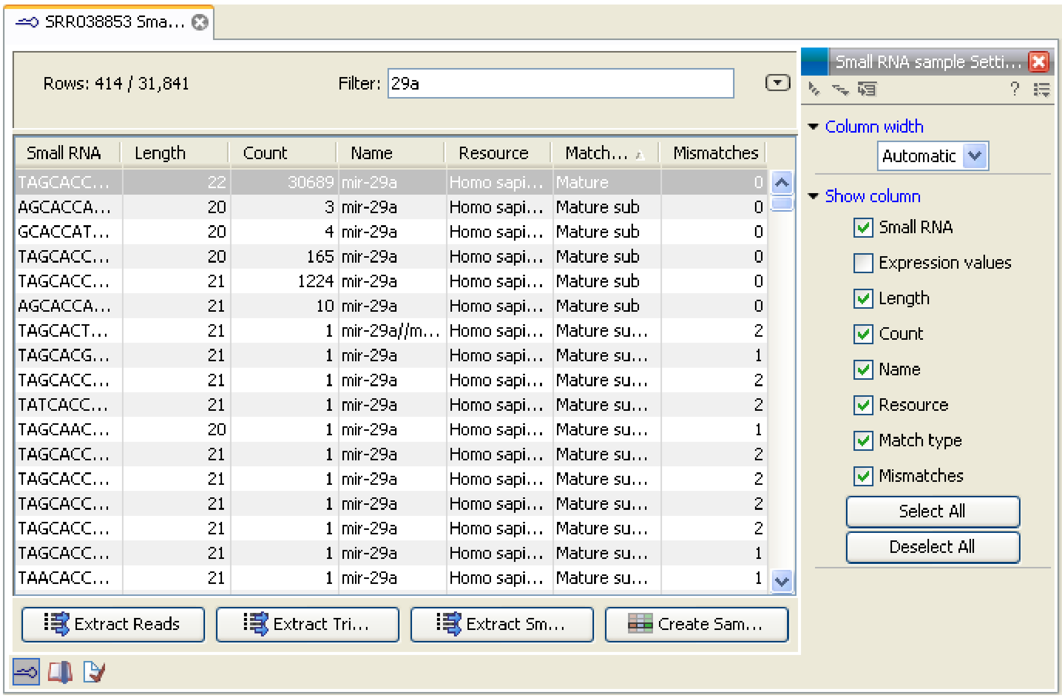
● Open

○ Save

Log handling

☐ Make log

Figure 2.95: *Select the sample grouped on mature.*



Rows: 414 / 31,841 Filter: 29a

Small RNA	Length	Count	Name	Resource	Match...	Mismatches
TAGCACC...	22	30689	mir-29a	Homo sapi...	Mature	0
AGCACCA...	20	3	mir-29a	Homo sapi...	Mature sub	0
GCACCAT...	20	4	mir-29a	Homo sapi...	Mature sub	0
TAGCACC...	20	165	mir-29a	Homo sapi...	Mature sub	0
TAGCACC...	21	1224	mir-29a	Homo sapi...	Mature sub	0
AGCACCA...	21	10	mir-29a	Homo sapi...	Mature sub	0
TAGCACT...	21	1	mir-29a//m...	Homo sapi...	Mature su...	2
TAGCACG...	21	1	mir-29a	Homo sapi...	Mature su...	1
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TATCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TAGCAAC...	20	1	mir-29a	Homo sapi...	Mature su...	1
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	2
TAGCACC...	21	1	mir-29a	Homo sapi...	Mature su...	1
TAACACC...	21	1	mir-29a	Homo sapi...	Mature su...	1

Small RNA sample Setti...

Column width: Automatic

Show column

☑ Small RNA

☐ Expression values

☑ Length

☑ Count

☑ Name

☑ Resource

☑ Match type

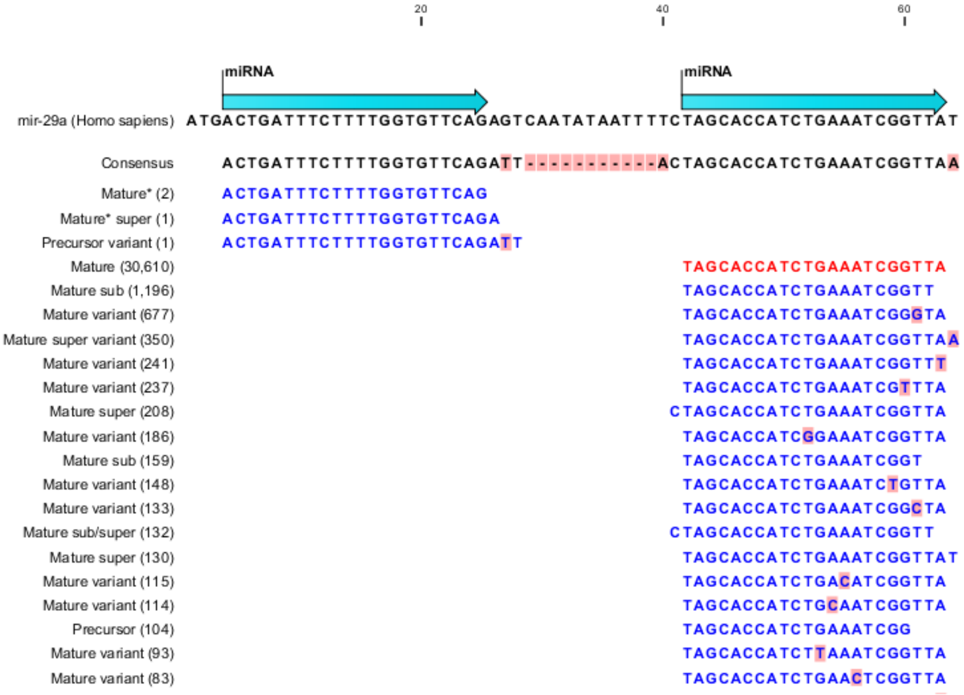
☑ Mismatches

Figure 2.96: *Showing all tags annotated with mir29a.*

out in a different color.

Taking advantage of the RNA folding opportunities

One of the advantages of *CLC Genomics Workbench* is the integration between various tools. We are now going to explore the RNA secondary structure prediction tool for this miRNA. First, right-click the mir-29a label in the mapping view and select **Open This Sequence**. This will open this sequence in a new view but it is still part of the mapping and the grouped sample (this is denoted by the square brackets around its name). We will now predict the secondary structure of



```
     mir-29a (Homo sapiens) ATGACTGATTTCTTTTGGTGTTCAGAGTCAATATAATTTTCTAGCACCATCTGAAATCGGTTAT

                  Consensus    ACTGATTTCTTTTGGTGTTCAGATT-----------ACTAGCACCATCTGAAATCGGTTAA
                Mature* (2)    ACTGATTTCTTTTGGTGTTCAG
          Mature* super (1)    ACTGATTTCTTTTGGTGTTCAGA
      Precursor variant (1)    ACTGATTTCTTTTGGTGTTCAGATT
            Mature (30,610)                                          TAGCACCATCTGAAATCGGTTA
         Mature sub (1,196)                                          TAGCACCATCTGAAATCGGTT
       Mature variant (677)                                          TAGCACCATCTGAAATCGGGTA
 Mature super variant (350)                                          TAGCACCATCTGAAATCGGTTAA
       Mature variant (241)                                          TAGCACCATCTGAAATCGGTTT
       Mature variant (237)                                          TAGCACCATCTGAAATCGTTTA
         Mature super (208)                                         CTAGCACCATCTGAAATCGGTTA
       Mature variant (186)                                          TAGCACCATCGGAAATCGGTTA
           Mature sub (159)                                          TAGCACCATCTGAAATCGGT
       Mature variant (148)                                          TAGCACCATCTGAAATCTGTTA
       Mature variant (133)                                          TAGCACCATCTGAAATCGGCTA
     Mature sub/super (132)                                         CTAGCACCATCTGAAATCGGTT
         Mature super (130)                                          TAGCACCATCTGAAATCGGTTAT
       Mature variant (115)                                          TAGCACCATCTGACATCGGTTA
       Mature variant (114)                                          TAGCACCATCTGCAATCGGTTA
            Precursor (104)                                          TAGCACCATCTGAAATCGG
        Mature variant (93)                                          TAGCACCATCTTAAATCGGTTA
        Mature variant (83)                                          TAGCACCATCTGAACTCGGTTA
```

Figure 2.97: *Showing the alignment against the mir29a precursor sequence.*

this sequence:

Toolbox | **RNA Structure** ( ) | **Predict Secondary Structure** ( )

Click **Next** and **Next** using the default settings, uncheck the option to add annotations and click **Finish**. Then switch to the **Secondary Structure 2D View** ( ) to see the predicted structure (see figure 2.98).

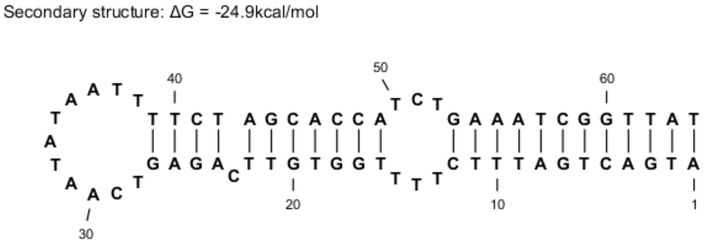


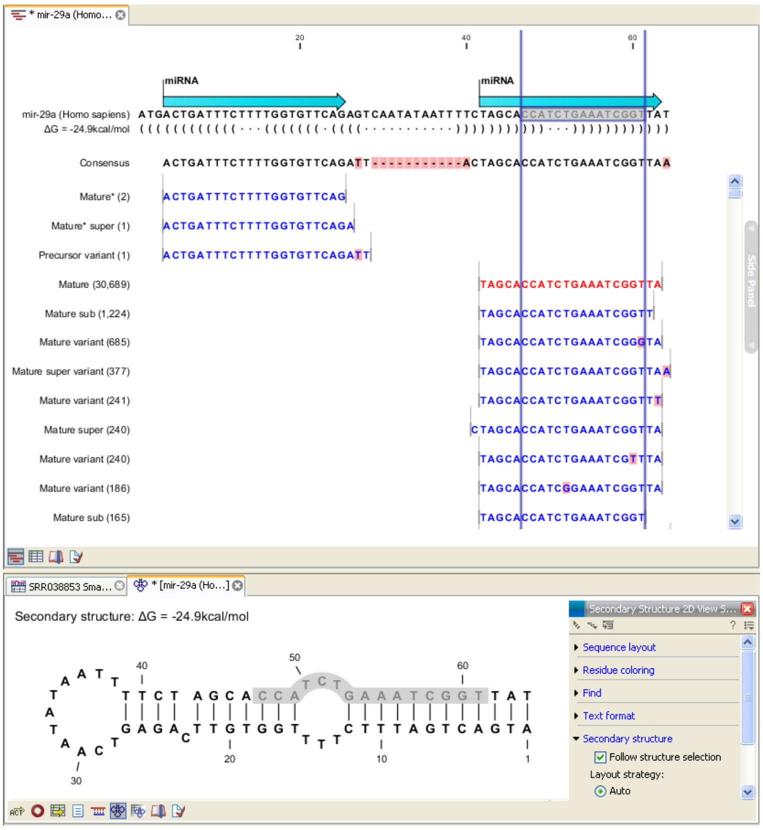
Figure 2.98: *Showing the folding of the mir-29a precursor.*

If your views are not already split, drag the tabs of the views to create a set-up as shown in figure 2.99 and select using the mouse either in the secondary structure or the reference in the mapping view and you will be able to follow the selections across the views.

Close the views. You are prompted to save the changes which in this case is the adding of the secondary structure to the precursor sequence.

Tracking back from the mature sample

Now, open the `SRR038853 Small RNA sample grouped on mature` ( ) and look at the first row with `let-7f-1//let-7f-2` in the **Name** column. The let-7f miRNA is annotated in



```
mir-29a (Homo sapiens)  ATGACTGATTTCTTTTGGTGTTCAGAGTCAATATAATTTTCTAGCACCATCTGAAATCGGTTAT
 ΔG = -24.9kcal/mol     (((((((((((···(((((((·((((···········))))))))))))···)))))))))))))

Consensus                 ACTGATTTCTTTTGGTGTTCAGATT-----------ACTAGCACCATCTGAAATCGGTTAA

Mature* (2)               ACTGATTTCTTTTGGTGTTCAG
Mature* super (1)         ACTGATTTCTTTTGGTGTTCAGA
Precursor variant (1)     ACTGATTTCTTTTGGTGTTCAGATT
Mature (30,689)                                                TAGCACCATCTGAAATCGGTTA
Mature sub (1,224)                                             TAGCACCATCTGAAATCGGTT
Mature variant (685)                                           TAGCACCATCTGAAATCGGGTA
Mature super variant (377)                                     TAGCACCATCTGAAATCGGTTAA
Mature variant (241)                                           TAGCACCATCTGAAATCGGTTT
Mature super (240)                                            CTAGCACCATCTGAAATCGGTTA
Mature variant (240)                                           TAGCACCATCTGAAATCGTTTA
Mature variant (186)                                           TAGCACCATCGGAAATCGGTTA
Mature sub (165)                                               TAGCACCATCTGAAATCGGT
```

Secondary structure: ΔG = -24.9kcal/mol

A A T T
T
A
T
A A C T
T T C T A G C A C C A T C T G A A A T C G G T T A T
G A G A C T T G T G G T T T T C T T T A G T C A G T A
40
50
60
30
20
10
1
Secondary Structure 2D View S...
Sequence layout
Residue coloring
Find
Text format
Secondary structure

☑ Follow structure selection

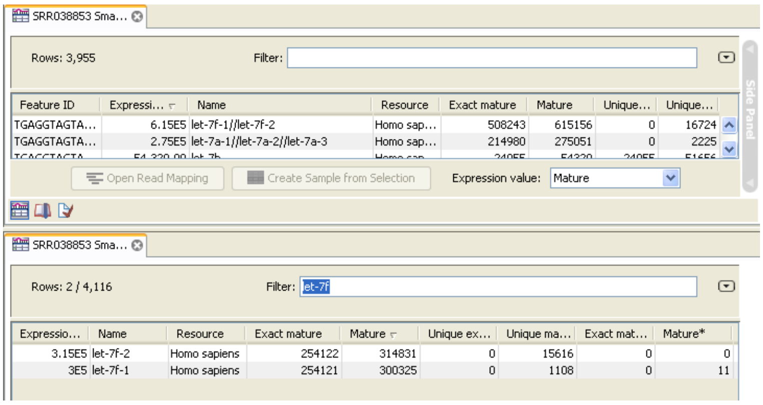
Layout strategy:

● Auto

Figure 2.99: *A split view showing the secondary structure of the RNA together with the length variants.*

miRBase with two different precursor sequences. This means that when the tags are annotated, they are assigned either to let-7f-1 or let-7f-2. The sample grouped on mature merges the tags from precursors sharing the same mature sequence (the sequence itself is shown in the **Feature id** column).

Open the `SRR038853 Small RNA sample grouped` and enter let-7f in the filter. The two precursor variants are now displayed, and you can see that the numbers sum up compared to the mature sample: 254,122 + 254,121 = 508,243 (see the **Mature** column in figure 2.100).



Rows: 3,955   Filter:

Feature ID	Expressi...	Name	Resource	Exact mature	Mature	Unique...	Unique...
TGAGGTAGTA...	6.15E5	let-7f-1//let-7f-2	Homo sap...	508243	615156	0	16724
TGAGGTAGTA...	2.75E5	let-7a-1//let-7a-2//let-7a-3	Homo sap...	214980	275051	0	2225
TCACCTACTA	[illegible]	let-7b	Homo sap...	[illegible]	[illegible]	[illegible]	[illegible]

Expression value: Mature

Rows: 2 / 4,116   Filter: let-7f

Expressio...	Name	Resource	Exact mature	Mature	Unique ex...	Unique ma...	Exact mat...	Mature*
3.15E5	let-7f-2	Homo sapiens	254122	314831	0	15616	0	0
3E5	let-7f-1	Homo sapiens	254121	300325	0	1108	0	11

Figure 2.100: *The mature sample joins the counts of precursors sharing the same mature sequence.*

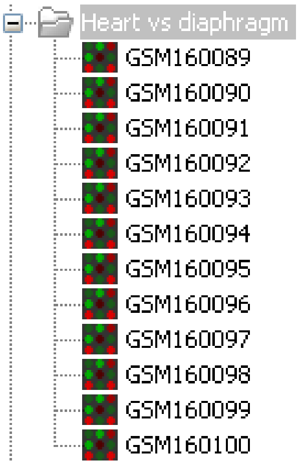
2.15 Tutorial: Microarray-based expression analysis part I: Getting started

This tutorial is the first part of a series of tutorials about expression analysis. Expression analysis often requires advanced skills in statistics, but this tutorial is intended to show a straight-forward example of how to identify and interpret the differentially expressed genes in samples from two different tissues. If you are familiar with the statistical concepts and issues within expression analysis, you may find this tutorial too simplistic, but we have favored a simple and quick introduction over an exhaustive and more "correct" explanation.

The data comes from a study of gene expression in tissues from cardiac left ventricle and diaphragm muscle of rats [van Lunteren et al., 2008]. During this series of tutorials, you will see how to import and set up the data in an experiment with two groups (part I), to perform quality checks on the data (part II), to perform statistics and clustering to identify and visualize differentially expressed genes (part III), and finally to use annotations to categorize and interpret patterns among the differentially expressed genes in a biological context (part IV).

2.15.1 Importing array data

First, import the data set which can be downloaded from the Gene Expression Omnibus (GEO) database at NCBI: `http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6943&targ=gsm&form=text&view=data`. After download, click **Import** () in the Tool bar and select the file. You will now have 12 arrays in your **Navigation Area** as shown in figure 2.101.



Heart vs diaphragm
GSM160089
GSM160090
GSM160091
GSM160092
GSM160093
GSM160094
GSM160095
GSM160096
GSM160097
GSM160098
GSM160099
GSM160100

Figure 2.101: *12 microarrays have been imported.*

2.15.2 Grouping the samples

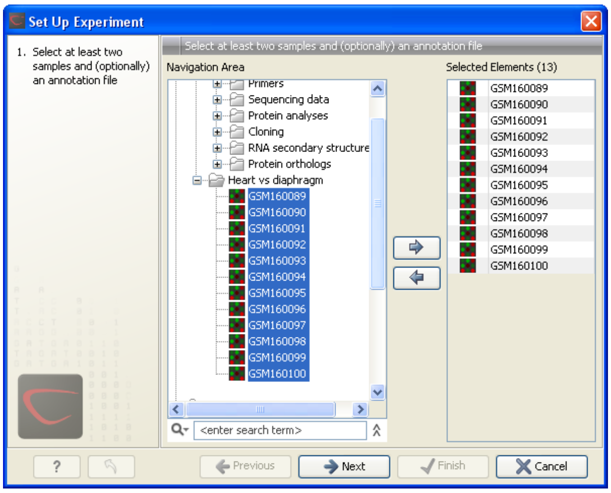
The next step is to tell the *CLC Genomics Workbench* how the 12 samples are related.

This is done be setting up an **Experiment** (). An **Experiment** is the central data type when analyzing expression data in the *CLC Genomics Workbench*. It includes a set of samples and information about how the samples are related (which groups they belong to). The **Experiment** is also used to accumulate calculations like t-tests and clustering.

First step is to set up the experiment:

Toolbox | **Expression Analysis (** **)** | **Set Up Experiment (** **)**

Select the 12 arrays that you have imported (see figure 2.102).



Set Up Experiment

1. Select at least two samples and (optionally) an annotation file

Select at least two samples and (optionally) an annotation file

Navigation Area

- Primers
- Sequencing data
- Protein analyses
- Cloning
- RNA secondary structure
- Protein orthologs
- Heart vs diaphragm
  - GSM160089
  - GSM160090
  - GSM160091
  - GSM160092
  - GSM160093
  - GSM160094
  - GSM160095
  - GSM160096
  - GSM160097
  - GSM160098
  - GSM160099
  - GSM160100

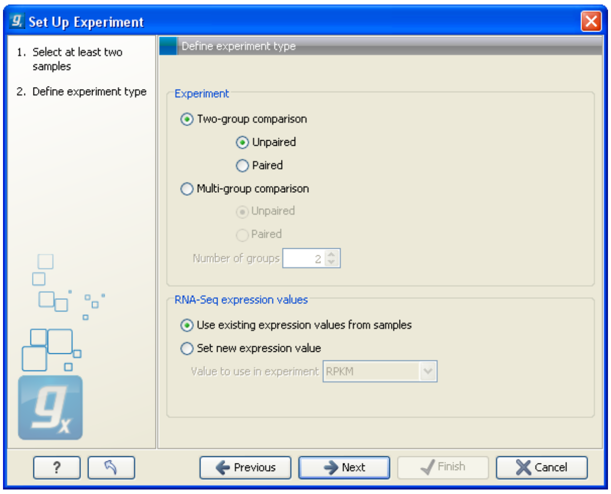
<enter search term>

Selected Elements (13)

- GSM160089
- GSM160090
- GSM160091
- GSM160092
- GSM160093
- GSM160094
- GSM160095
- GSM160096
- GSM160097
- GSM160098
- GSM160099
- GSM160100

Figure 2.102: *Select the 12 microarrays that have been imported.*

Note that we use "samples" as the general term for both microarray-based expression values and sequencing-based expression values. Clicking **Next** shows the dialog in figure 2.103.



Set Up Experiment

1. Select at least two samples
2. Define experiment type

Define experiment type

Experiment

● Two-group comparison

● Unpaired

○ Paired

○ Multi-group comparison

● Unpaired

○ Paired

Number of groups 2

RNA-Seq expression values

● Use existing expression values from samples

○ Set new expression value

Value to use in experiment RPKM

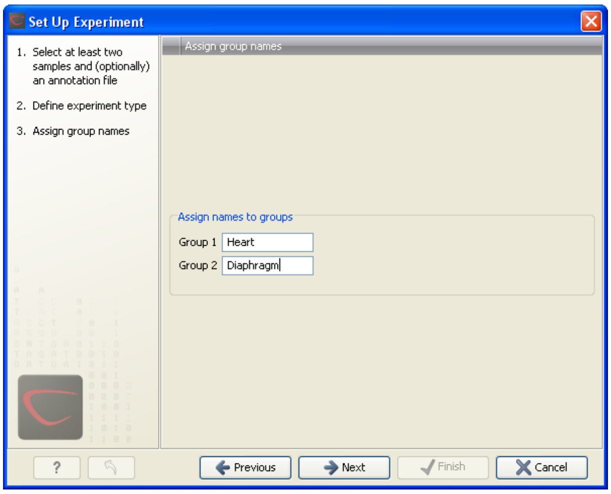
Figure 2.103: *Defining the number of groups.*

Here you define the number of groups in the experiment. Since we compare heart tissue with diaphragm tissue, we use a two-group comparison. Leave it as **Unpaired**. Clicking **Next** shows the dialog in figure 2.104.

Name the first group **Heart** and the second group **Diaphragm** and click **Next** (see figure 2.105).

Here you see a list of all the samples you chose in figure 2.102. Now select the first 6 samples (by clicking in the group column of the first sample and while holding down the mouse button you drag and select the other five samples), right-click and select **Heart**. Select the last 6 samples, right-click and select **Diaphragm**. In this way you define which group each sample belongs to.

Click **Finish** and the experiment will be created. Note that the information from samples located



Set Up Experiment

1. Select at least two samples and (optionally) an annotation file
2. Define experiment type
3. Assign group names

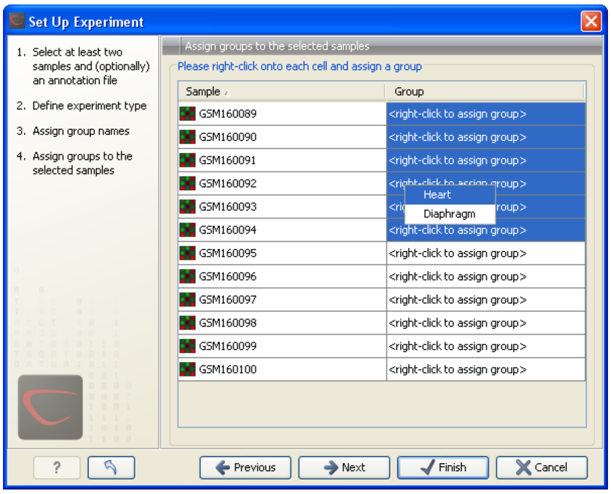
Assign group names

Assign names to groups

Group 1 Heart

Group 2 Diaphragm

Figure 2.104: *Naming the groups.*



Set Up Experiment

1. Select at least two samples and (optionally) an annotation file
2. Define experiment type
3. Assign group names
4. Assign groups to the selected samples

Assign groups to the selected samples

Please right-click onto each cell and assign a group

Sample	Group
GSM160089	<right-click to assign group>
GSM160090	<right-click to assign group>
GSM160091	<right-click to assign group>
GSM160092	<right-click to assign group>
GSM160093	<ri roup>
GSM160094	<right-click to assign group>
GSM160095	<right-click to assign group>
GSM160096	<right-click to assign group>
GSM160097	<right-click to assign group>
GSM160098	<right-click to assign group>
GSM160099	<right-click to assign group>
GSM160100	<right-click to assign group>

Figure 2.105: *Assigning the samples to groups.*

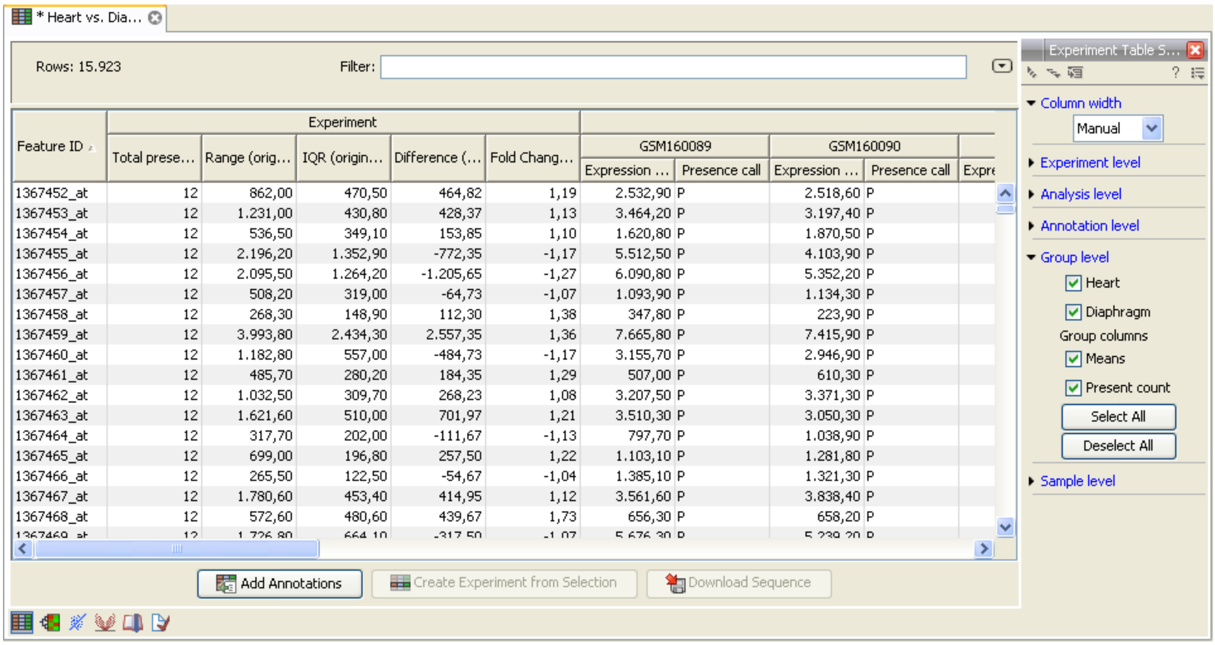
in the **Navigation Area** is copied into the experiment, so they now exist independently of each other.

2.15.3 The experiment table

Once it is created, the experiment will be opened in a table as shown in figure 2.106.

The table includes the expression values for each sample and in addition a few extra values have been calculated such as the range, the IQR (Interquartile Range), fold change and difference values and the present counts for the whole experiment and the individual groups (note that absent/present calls are not available on all kinds of data).

Save the experiment and you are ready to proceed to the expression analysis tutorial part II.



Rows: 15.923

Filter:

Feature ID	Experiment					GSM160089		GSM160090		
	Total prese...	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Expression ...	Presence call	Expression ...	Presence call	Expre
1367452_at	12	862,00	470,50	464,82	1,19	2.532,90	P	2.518,60	P	
1367453_at	12	1.231,00	430,80	428,37	1,13	3.464,20	P	3.197,40	P	
1367454_at	12	536,50	349,10	153,85	1,10	1.620,80	P	1.870,50	P	
1367455_at	12	2.196,20	1.352,90	-772,35	-1,17	5.512,50	P	4.103,90	P	
1367456_at	12	2.095,50	1.264,20	-1.205,65	-1,27	6.090,80	P	5.352,20	P	
1367457_at	12	508,20	319,00	-64,73	-1,07	1.093,90	P	1.134,30	P	
1367458_at	12	268,30	148,90	112,30	1,38	347,80	P	223,90	P	
1367459_at	12	3.993,80	2.434,30	2.557,35	1,36	7.665,80	P	7.415,90	P	
1367460_at	12	1.182,80	557,00	-484,73	-1,17	3.155,70	P	2.946,90	P	
1367461_at	12	485,70	280,20	184,35	1,29	507,00	P	610,30	P	
1367462_at	12	1.032,50	309,70	268,23	1,08	3.207,50	P	3.371,30	P	
1367463_at	12	1.621,60	510,00	701,97	1,21	3.510,30	P	3.050,30	P	
1367464_at	12	317,70	202,00	-111,67	-1,13	797,70	P	1.038,90	P	
1367465_at	12	699,00	196,80	257,50	1,22	1.103,10	P	1.281,80	P	
1367466_at	12	265,50	122,50	-54,67	-1,04	1.385,10	P	1.321,30	P	
1367467_at	12	1.780,60	453,40	414,95	1,12	3.561,60	P	3.838,40	P	
1367468_at	12	572,60	480,60	439,67	1,73	656,30	P	658,20	P	
1367469_at	12	1.726,80	664,10	-317,50	-1,07	5.676,30	P	5.239,20	P	

- Column width: Manual
- Experiment level
- Analysis level
- Annotation level
- Group level
  - ☑ Heart
  - ☑ Diaphragm
  - Group columns
  - ☑ Means
  - ☑ Present count
  - Select All
  - Deselect All
- Sample level

Figure 2.106: *The experiment table.*

2.16 Tutorial: Microarray-based expression analysis part II: Quality control

This tutorial is the second part of a series of tutorials about expression analysis. We continue working with the data set introduced in the first tutorial.

In this tutorial we will examine various methods to perform quality control of the data.

2.16.1 Transformation

First we inspect to what extent the variance in expression values depends on the mean. For this we create an **MA Plot**:

> **Toolbox** | **Expression Analysis (** **)** | **General Plots** | **Create MA Plot (** **)**

Since the MA plot compares two samples, select two of the 12 arrays and click **Finish**. This will show a plot similar to the one shown in figure 2.107.

The X axis shows the mean expression level of a feature on the two arrays and the Y axis shows the difference in expression levels for a feature on the two arrays. From the plot shown in figure 20.59 it is clear that the variance increases with the mean. To remove some of the dependency, we want to **transform** the data:

> **Toolbox** | **Expression Analysis (** **)** | **Transformation and Normalization** | **Transform (** **)**

Select the same arrays used for the plot, click **Next**, choose **Log 2** transformation and click **Finish**. Now create an MA plot again as described above, but when you click **Next** you can see that you now also have the option to choose **Transformed expression values** (see figure 2.108).

Select the transformed values. You will see that these three selection boxes; Original, Trans-

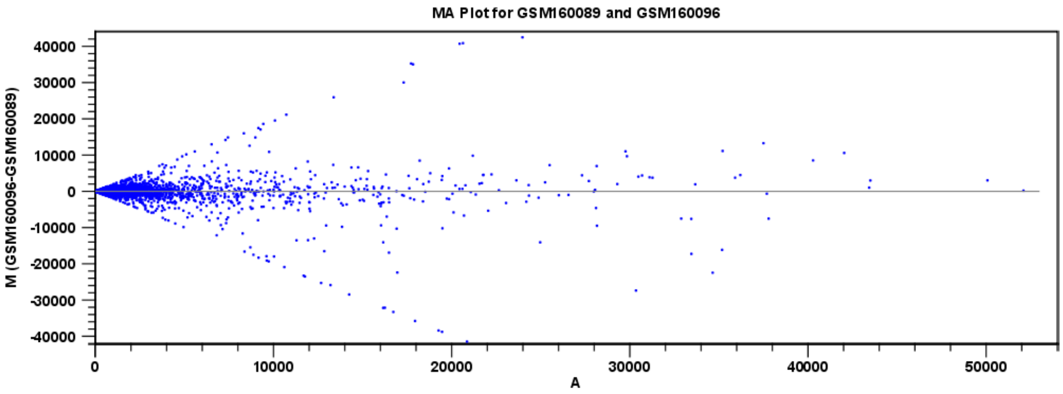
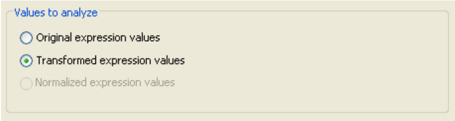


Figure 2.107: *MA plot before transformation.*



Values to analyze

○ Original expression values
● Transformed expression values
○ Normalized expression values

Figure 2.108: *Select the transformed expression values.*

formed and Normalized expression values are used several places when expression values are used in a calculation.

Click **Finish**.

This will result in a quite different plot as shown in figure 2.109.

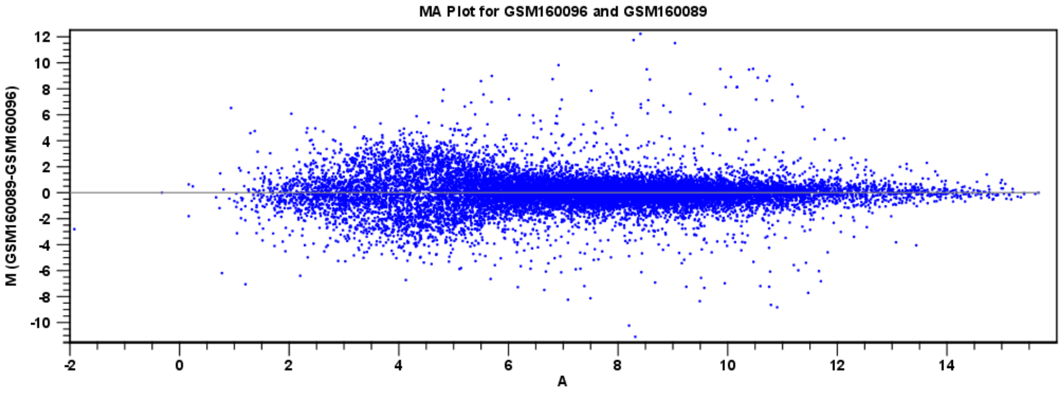
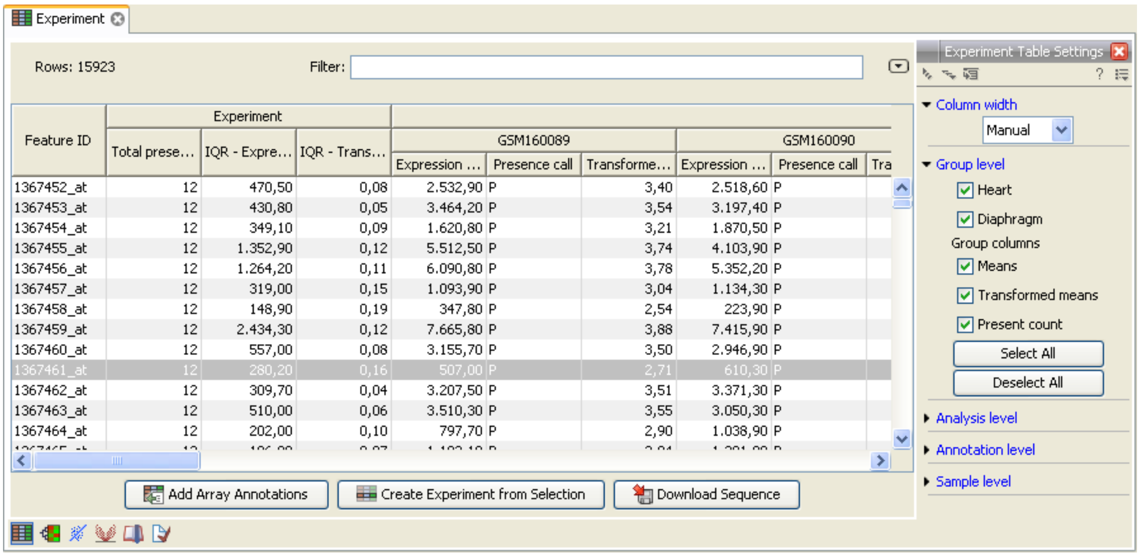


Figure 2.109: *MA plot after transformation.*

The much more symmetric and even spread indicates that the dependance of the variance on the mean is not as strong as it was before transformation.

We have now only transformed the values of the two samples used for the MA plot. The next step is to transform the expression values within the experiment, since this is the data we are going to use in the further analysis. The procedure is similar to before - this time you just select the experiment created in the first part of this tutorial series instead of the two arrays.

If you open the table, you will see that all the samples have an extra column with transformed expression values (see figure 2.110).



Experiment

Rows: 15923 Filter:

Feature ID	Experiment									
				GSM160089			GSM160090			
	Total prese...	IQR - Expre...	IQR - Trans...	Expression ...	Presence call	Transforme...	Expression ...	Presence call	Tra
1367452_at	12	470,50	0,08	2.532,90	P	3,40	2.518,60	P	
1367453_at	12	430,80	0,05	3.464,20	P	3,54	3.197,40	P	
1367454_at	12	349,10	0,09	1.620,80	P	3,21	1.870,50	P	
1367455_at	12	1.352,90	0,12	5.512,50	P	3,74	4.103,90	P	
1367456_at	12	1.264,20	0,11	6.090,80	P	3,78	5.352,20	P	
1367457_at	12	319,00	0,15	1.093,90	P	3,04	1.134,30	P	
1367458_at	12	148,90	0,19	347,80	P	2,54	223,90	P	
1367459_at	12	2.434,30	0,12	7.665,80	P	3,88	7.415,90	P	
1367460_at	12	557,00	0,08	3.155,70	P	3,50	2.946,90	P	
1367461_at	12	280,20	0,16	507,00	P	2,71	610,30	P	
1367462_at	12	309,70	0,04	3.207,50	P	3,51	3.371,30	P	
1367463_at	12	510,00	0,06	3.510,30	P	3,55	3.050,30	P	
1367464_at	12	202,00	0,10	797,70	P	2,90	1.038,90	P	

Experiment Table Settings

Column width: Manual

Group level
☑ Heart
☑ Diaphragm

Group columns
☑ Means
☑ Transformed means
☑ Present count

Figure 2.110: *Transformed expression values have been added to the table.*

There is also an extra column for transformed group means and transformed IQR.

2.16.2 Comparing spread and distribution

In order to perform meaningful statistical analysis and inferences from the data, you need to ensure that the samples are comparable. Systematic differences between the samples that are likely to be due to noise (e.g. differences in sample preparation and processing) rather than true biological variability should be removed. To examine and compare the overall distribution of the transformed expression values in the samples you may use a **Box plot** (⌸):

Toolbox | **Expression Analysis (**⌸**)** | **Quality Control** | **Create Box Plot (**⌸**)**

Select the experiment and click **Next**. Choose the **Transformed expression values** and **Finish**.

The box plot is shown in figure 2.111.

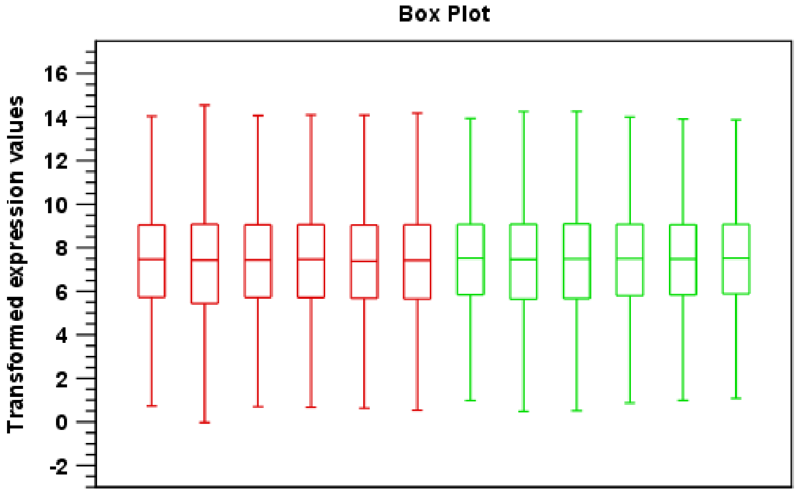


Figure 2.111: *A box plot of the 12 samples in the experiment, colored by group.*

This plot looks very good because none of the samples stands out from the rest. If you compare this plot to the one shown in figure 2.112 from another data set, you can see the difference.

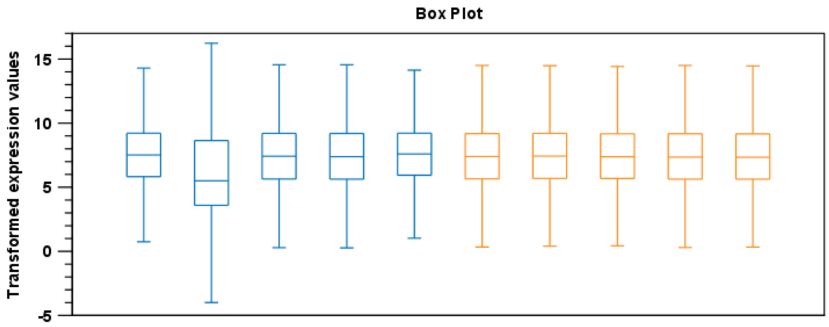


Figure 2.112: *A box plot showing one sample that stands out from the rest.*

The second sample from the left has a distribution that is quite different from the others. If you have a data set like this, then you should consider removing the bad quality sample.

2.16.3 Group differentiation

The next step in the quality control is to check whether the overall variability of the samples reflect their grouping. In other words we want the replicates to be relatively homogenous and distinguishable from the samples of the other group.

First, we perform a **Principal Component Analysis (PCA)**:

> **Toolbox** | **Expression Analysis** (📁) | **Quality Control** | **Principal Component Analysis** (📊)

Select the experiment and click **Next**. **Finish**. This will create a PCA plot as shown in figure 2.113).

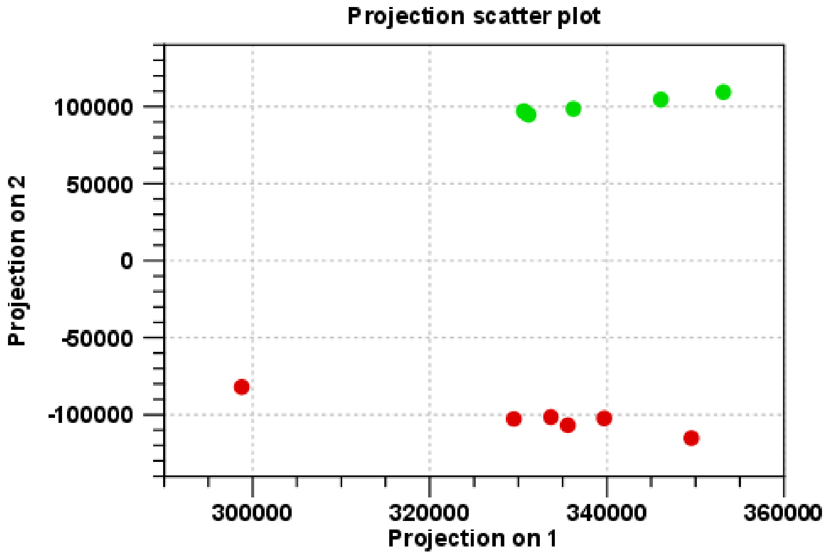


Figure 2.113: *A principal component analysis colored by group.*

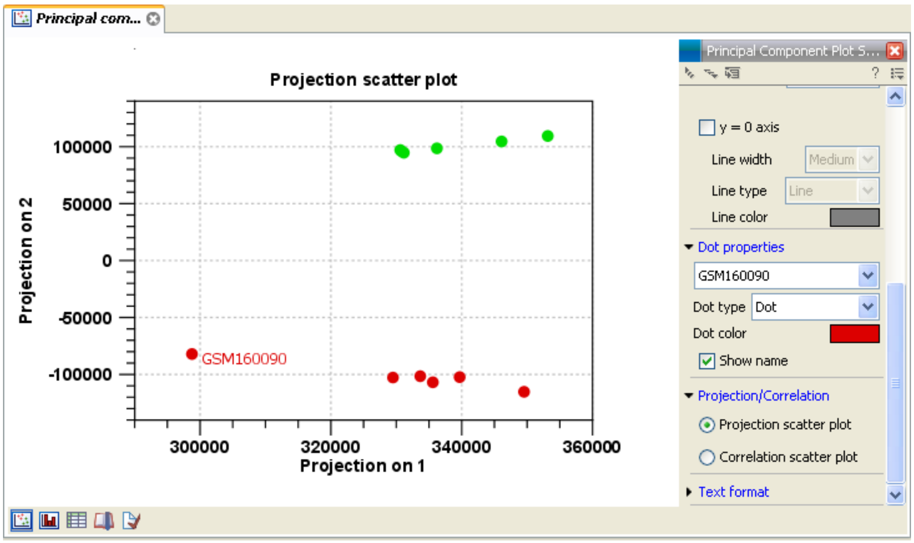
The plot shows the projection of the samples onto the two-dimensional space spanned by the first and second principal component. (These are the orthogonal directions in which the data exhibits the largest and second-largest variability).

The dots are colored according to the groups, and they also group very nicely in the plot. There is only one outlier - to see which sample it is, place the mouse cursor on the dot for a second, and you will see that it is the *GSM160090* from the *Heart* group.

You can display this information in the plot using the settings in the **Side Panel** to the right of the view:

> **Dot properties | select GSM160090 in the drop-down box | Show names**

In this way you can control the coloring and dot types of the different samples and groups (see figure 2.114).



□ y = 0 axis

Line width Medium

Line type Line

Line color

GSM160090

Dot type Dot

Dot color

☑ Show name

● Projection scatter plot

○ Correlation scatter plot

Figure 2.114: *Naming the outlier.*

In order to complement the principal component analysis, we will also do a hierarchical clustering of the samples to see if the samples cluster in the groups we expect:

> **Toolbox | Expression Analysis ( ) | Quality Control | Hierarchical Clustering of Samples ( )**

Select the experiment and click **Next**. Leave the parameters at their default and click **Finish**.

This will display a heat map showing the clustering of samples at the bottom (see figure 2.115).

The two overall groups formed are identical to the grouping in the experiment. You can double-check by placing your mouse on the name of the sample - that will show which group it belongs to.

Since both the principal component analysis and the hierarchical clustering confirms the grouping of the samples, we have no reason to be sceptical about the quality of the samples and we conclude that the data is OK.

Note that the heat map is not a new element to be stored in the **Navigation Area** - it is just another way of looking at the experiment (note the buttons to switch between different views in

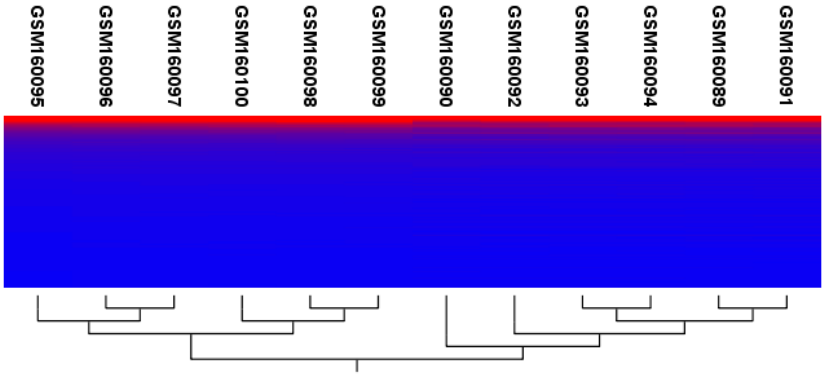


Figure 2.115: *Sample clustering.*

figure 2.116.

Figure 2.116: *Different views on an experiment.*

In part III of the tutorial series we will be looking into the different views in more detail.

To summarize this part about quality control, it looks like the data have good quality, and we are now ready to proceed to the next step where we do some statistical analysis to see which genes are differentially expressed.

2.17 Tutorial: Microarray-based expression analysis part III: Differentially expressed genes

This tutorial is the third part of a series of tutorials about expression analysis. We continue working with the data set introduced in the first tutorial.

In this tutorial we will identify and investigate the genes that are differentially expressed.

2.17.1 Statistical analysis

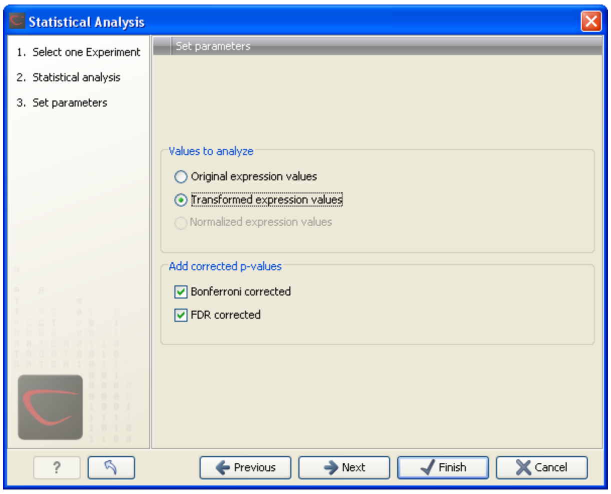
First we will carry out some statistical tests that we will use to identify the genes that are differentially expressed between the two groups:

Toolbox | **Expression Analysis** ( ) | **Statistical Analysis** | **On Gaussian Data** ( )

Select the experiment created in part I of the tutorials and click **Next**. Leave the parameters at the default and click **Next** again. You will now see a dialog as shown in figure 2.117.

As shown in figure 2.117 select the transformed expression values and check the two corrected p-values as well. You can read more about what they mean by clicking the **Help** (?) button in the dialog.

When you press **Finish**, a number of extra columns will be added to your experiment. For this



Statistical Analysis

1. Select one Experiment
2. Statistical analysis
3. Set parameters

Set parameters

Values to analyze

○ Original expression values
● Transformed expression values
○ Normalized expression values

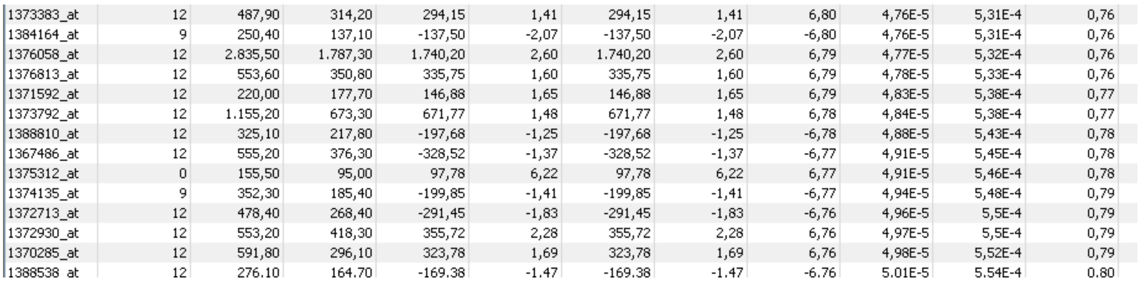
Add corrected p-values

☑ Bonferroni corrected
☑ FDR corrected

Figure 2.117: *Statistical analysis.*

analysis we will use the FDR p-value which is a measure that allows us to control how big a proportion of false positives (genes that we think are differentially expressed but really are not) we are willing to accept.

Click the **FDR p-value correction** column to sort it with the lowest values at the top. If you scroll down to values around 5E-4 you can clearly see the difference between using the FDR p-value and the Bonferroni-corrected p-value which is much stricter (p-values approaching 1 - see figure 2.118).



1373383_at	12	487,90	314,20	294,15	1,41	294,15	1,41	6,80	4,76E-5	5,31E-4	0,76
1384164_at	9	250,40	137,10	-137,50	-2,07	-137,50	-2,07	-6,80	4,76E-5	5,31E-4	0,76
1376058_at	12	2.835,50	1.787,30	1.740,20	2,60	1.740,20	2,60	6,79	4,77E-5	5,32E-4	0,76
1376813_at	12	553,60	350,80	335,75	1,60	335,75	1,60	6,79	4,78E-5	5,33E-4	0,76
1371592_at	12	220,00	177,70	146,88	1,65	146,88	1,65	6,79	4,83E-5	5,38E-4	0,77
1373792_at	12	1.155,20	673,30	671,77	1,48	671,77	1,48	6,78	4,84E-5	5,38E-4	0,77
1388810_at	12	325,10	217,80	-197,68	-1,25	-197,68	-1,25	-6,78	4,88E-5	5,43E-4	0,78
1367486_at	12	555,20	376,30	-328,52	-1,37	-328,52	-1,37	-6,77	4,91E-5	5,45E-4	0,78
1375312_at	0	155,50	95,00	97,78	6,22	97,78	6,22	6,77	4,91E-5	5,46E-4	0,78
1374135_at	9	352,30	185,40	-199,85	-1,41	-199,85	-1,41	-6,77	4,94E-5	5,48E-4	0,79
1372713_at	12	478,40	268,40	-291,45	-1,83	-291,45	-1,83	-6,76	4,96E-5	5,5E-4	0,79
1372930_at	12	553,20	418,30	355,72	2,28	355,72	2,28	6,76	4,97E-5	5,5E-4	0,79
1370285_at	12	591,80	296,10	323,78	1,69	323,78	1,69	6,76	4,98E-5	5,52E-4	0,79
1388538 at	12	276.10	164.70	-169.38	-1.47	-169.38	-1.47	-6.76	5.01E-5	5.54E-4	0.80

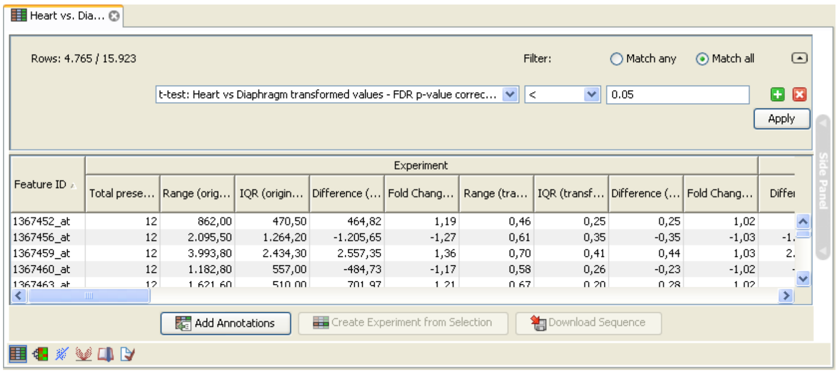
Figure 2.118: *FDR p-values compared to Bonferroni-corrected p-values.*

2.17.2 Filtering p-values

To do a more refined selection of the genes that we believe to be differentially expressed, we use the advanced filter located at the top of the experiment table. Click the **Advanced Filter** (▾) button and you will see that the simple text-based filter is now replaced with a more advanced filter. Select **Diaphragm vs Heart transformed - FDR p-value correction** in the first drop-down box, select **<** in the next, and enter 0.0005 (or 0,0005 depending on your locale settings). Click **Apply** or press **Enter**.

This will filter the table so that only values below 0.0005 are shown (see figure 2.119).

You can see that 1471 genes fulfilled this criterion (marked with a red circle).



Heart vs. Dia...

Rows: 4.765 / 15.923

Filter: ○ Match any ● Match all

t-test: Heart vs Diaphragm transformed values - FDR p-value correc... | < | 0.05

Apply

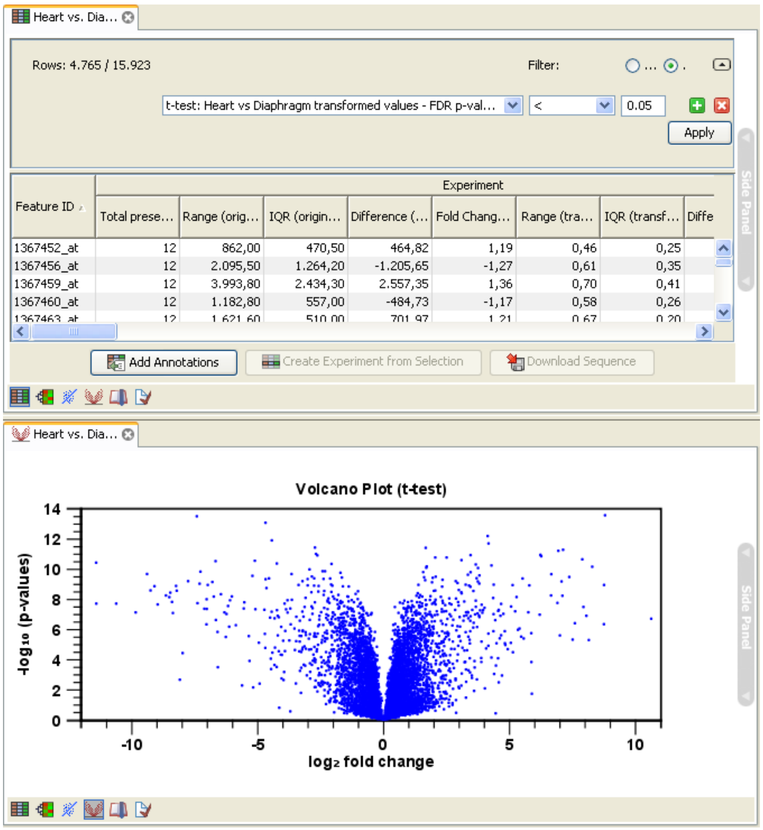
Feature ID	Experiment									
	Total prese...	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Range (tra...	IQR (transf...	Difference (...	Fold Chang...	Differ
1367452_at	12	862,00	470,50	464,82	1,19	0,46	0,25	0,25	1,02	
1367456_at	12	2.095,50	1.264,20	-1.205,65	-1,27	0,61	0,35	-0,35	-1,03	-1.
1367459_at	12	3.993,80	2.434,30	2.557,35	1,36	0,70	0,41	0,44	1,03	2.
1367460_at	12	1.182,80	557,00	-484,73	-1,17	0,58	0,26	-0,23	-1,02	-
1367463_at	12	1.621,60	510,00	701,97	1,21	0,67	0,20	0,28	1,02	

Figure 2.119: *Filtering on FDR p-values.*

2.17.3 Inspecting the volcano plot

Another way of looking at this data is to click the **Volcano Plot** (🌋) at the bottom of the view. Press and hold the Ctrl key while you click (⌘ on Mac).

This will make a split view of the experiment table and the volcano plot as shown in figure 2.120.



Heart vs. Dia...

Rows: 4.765 / 15.923

Filter: ○ ... ● .

t-test: Heart vs Diaphragm transformed values - FDR p-val... | < | 0.05

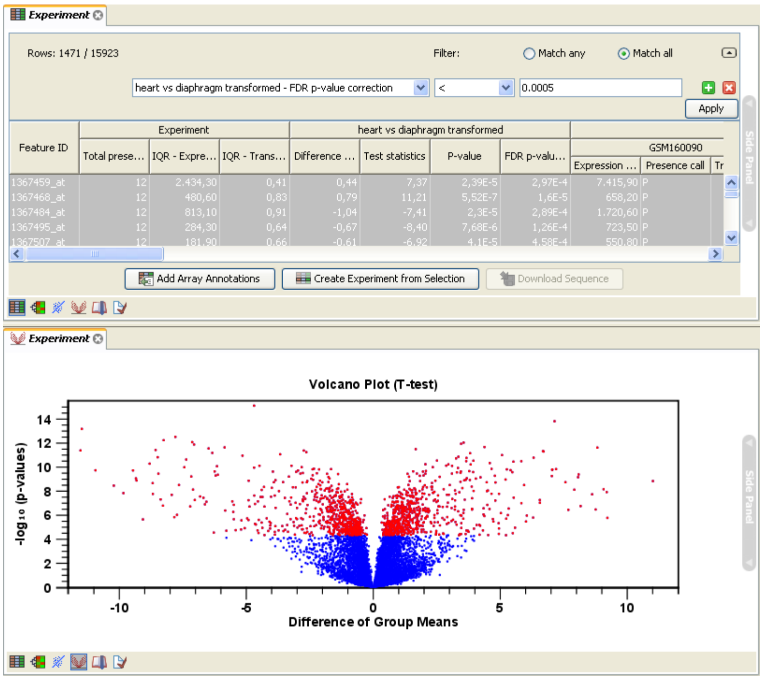
Apply

Feature ID	Experiment							
	Total prese...	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Range (tra...	IQR (transf...	Diffe
1367452_at	12	862,00	470,50	464,82	1,19	0,46	0,25	
1367456_at	12	2.095,50	1.264,20	-1.205,65	-1,27	0,61	0,35	
1367459_at	12	3.993,80	2.434,30	2.557,35	1,36	0,70	0,41	
1367460_at	12	1.182,80	557,00	-484,73	-1,17	0,58	0,26	
1367463_at	12	1.621,60	510,00	701,97	1,21	0,67	0,20	

Figure 2.120: *Split view of volcano plot and experiment table.*

The volcano plot shows the difference between the means of the two groups on the X axis and the $-\log_{10}$ p-values on the Y axis.

If you now select the genes in the table (click in the table and press Ctrl + A /⌘ + A on Mac), you can see that the corresponding dots are selected in the volcano plot (see figure 2.121).



Experiment

Rows: 1471 / 15923   Filter: ○ Match any ● Match all

heart vs diaphragm transformed - FDR p-value correction | < | 0.0005

Apply

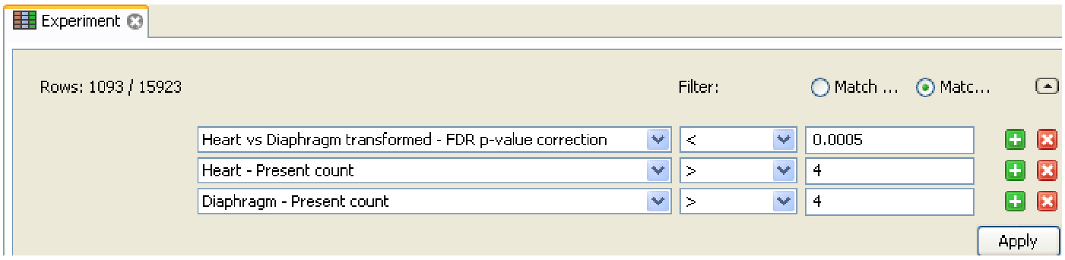
Feature ID	Experiment			heart vs diaphragm transformed				GSM160090		
	Total prese...	IQR - Expre...	IQR - Trans...	Difference ...	Test statistics	P-value	FDR p-valu...	Expression ...	Presence call	Tr
1367459_at	12	2.434,30	0,41	0,44	7,37	2,39E-5	2,97E-4	7.415,90	P	
1367468_at	12	480,60	0,83	0,79	11,21	5,52E-7	1,6E-5	658,20	P	
1367484_at	12	813,10	0,91	-1,04	-7,41	2,3E-5	2,89E-4	1.720,60	P	
1367495_at	12	284,30	0,64	-0,67	-8,40	7,68E-6	1,26E-4	723,50	P	
1367507_at	12	181,90	0,66	-0,61	-6,92	4,1E-5	4,58E-4	550,80	P	

Figure 2.121: *Volcano plot where selected dots are colored red.*

2.17.4 Filtering absent/present calls and fold change

Besides filtering on low p-values, we may also take the absent/present status of the features into consideration. The absent/present status is assigned by the Affymetrix software. There can be a number of reasons why a gene is called *absent*, and sometimes it is simply because the signal is very weak. When a gene is called absent, we may not wish to include it in the list of differentially expressed genes, so we want to filter these out as well.

This can be done in several ways - in our approach we say that for any gene there must not be more than one absent call in each group. Thus, we add more criteria to the filter by clicking the **Add search criterion** (+) button twice and enter the limit for present calls as shown in figure 2.122.



Experiment

Rows: 1093 / 15923   Filter: ○ Match ... ● Matc...

Heart vs Diaphragm transformed - FDR p-value correction	<	0.0005
Heart - Present count	>	4
Diaphragm - Present count	>	4

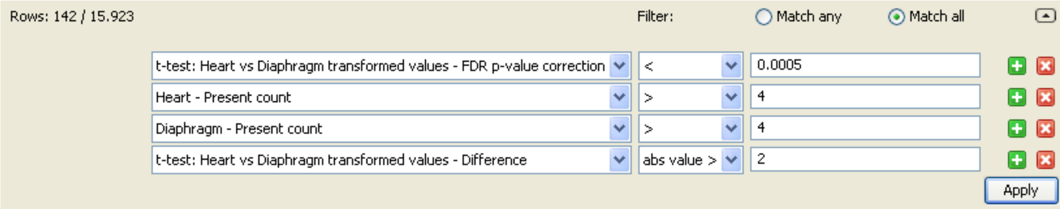
Apply

Figure 2.122: *Filtering genes where at least 5 out of 6 calls in each group are present.*

Before applying this filter, 1471 genes were selected, and this list is now reduced to 1093.

Often the results of microarray experiments are verified using other methods such as QPCR, and then we may want to filter out genes that exhibit differences in expression that are so small that we will not be able to verify them with another method. This is done by adding one last criterion to the filter: Difference should have an absolute value higher than 2 (as we are working with log transformed data, the group mean difference is really the *fold change*, so this filter means that we require a fold change above 2).

This final filtering is shown in figure 2.123.



Rows: 142 / 15.923

Filter: ○ Match any ● Match all

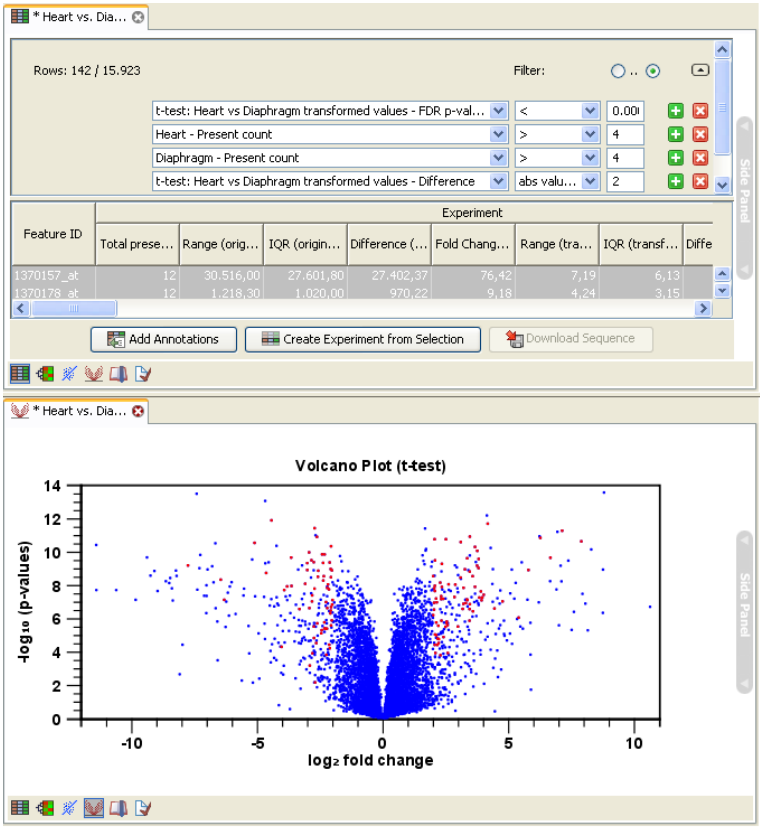
t-test: Heart vs Diaphragm transformed values - FDR p-value correction	<	0.0005
Heart - Present count	>	4
Diaphragm - Present count	>	4
t-test: Heart vs Diaphragm transformed values - Difference	abs value >	2

Figure 2.123: *The absolute value of group mean difference should be larger than 2.*

Note that the **abs value >** is important because the difference could be negative as well as positive.

The result is that we end up with a list of 142 genes that are likely candidates to exhibit differential expression in the two groups.

Click one of the rows and press Ctrl + A (⌘ +A on Mac) to select the 142 genes. You can now inspect the selection in the volcano plot below as shown in figure 2.124.



Rows: 142 / 15.923

Filter: ○ .. ●

t-test: Heart vs Diaphragm transformed values - FDR p-val...	<	0.00(
Heart - Present count	>	4
Diaphragm - Present count	>	4
t-test: Heart vs Diaphragm transformed values - Difference	abs valu...	2

Feature ID	Experiment							
	Total prese...	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Range (tra...	IQR (transf...	Diffe
1370157_at	12	30.516,00	27.601,80	27.402,37	76,42	7,19	6,13	
1370178_at	12	1.218,30	1.020,00	970,22	9,18	4,24	3,15	

Figure 2.124: *142 genes out of 15923 selected.*

2.17.5 Saving the gene list

Before we proceed to the final part of the tutorials, we save the list of genes; click **Create Experiment from Selection (▦)**

This will create a new experiment based on the selection. **Save** (💾) the new experiment next to the old one.

2.18 Tutorial: Microarray-based expression analysis part IV: Annotation test

This tutorial is the fourth and final part of a series of tutorials about expression analysis. We continue working with the data set introduced in the first tutorial and analyzed in part two and three.

In this tutorial we will annotate the gene list and use the annotations to see if there is a pattern in the biological annotations of the genes in the list of candidate differentially expressed genes.

We use two different methods for annotation testing: Hypergeometric Tests on Annotations and Gene Set Enrichment Analysis (GSEA).

2.18.1 Importing and adding the annotations

First step is to import an annotation file used to annotate the arrays. In this case, the data were produced using an Affymetrix chip, and the annotation file can be downloaded from the web site http://www.affymetrix.com/support/technical/annotationfilesmain.affx. You can access the file by search for **RAE230A**. Note that you have to sign up in order to download the file (this is a free service).

To import the annotation file, click **Import** ( ) in the Tool bar and select the file.

Next, annotate the experiment with the annotation file:

> **Toolbox | Expression Analysis ( ) | Annotation Test | Add Annotations ( )**

Select the experiment created in the previous tutorial and the annotation file ( ) and click **Next** and **Finish**.

2.18.2 Inspecting the annotations

When you look in the **Side Panel** of the experiment, there are a lot of options to show and hide columns in the table. This can be done on several levels. At the **Annotation level** you find a list of all the annotations. Some are shown per default, others you will have to click to show.

An important annotation is the **Gene title** which describes the gene and is much more informative than the Feature ID.

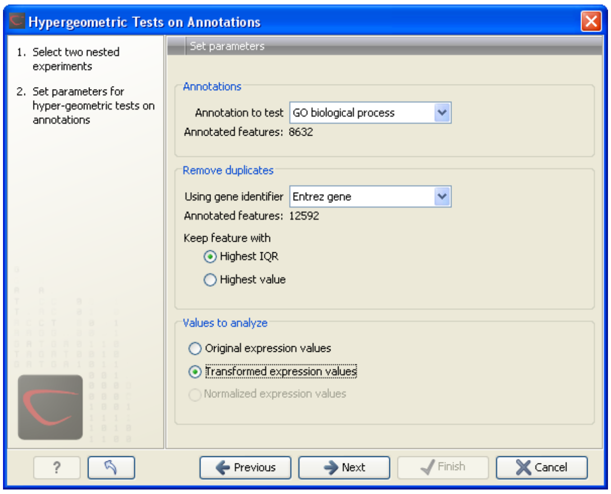
Further down the list you find the annotation type **GO biological process**. We will use this annotation in the next two analyses.

2.18.3 Processes that are over or under represented in the small list

The first annotation test will show whether any of the GO biological processes are over-represented in our small list of 142 differentially expressed genes relate to the full set of genes measured:

> **Toolbox | Expression Analysis ( ) | Annotation Test | Hypergeometric Tests on Annotations ( )**

Select the two experiments (the original full experiment and the small subset of 142 genes) and click **Next**. Select **GO biological process** and **Transformed expression values** (see figure 2.125).



Hypergeometric Tests on Annotations

1. Select two nested experiments
2. Set parameters for hyper-geometric tests on annotations

Set parameters

Annotations

Annotation to test: GO biological process

Annotated features: 8632

Remove duplicates

Using gene identifier: Entrez gene

Annotated features: 12592

Keep feature with

● Highest IQR

○ Highest value

Values to analyze

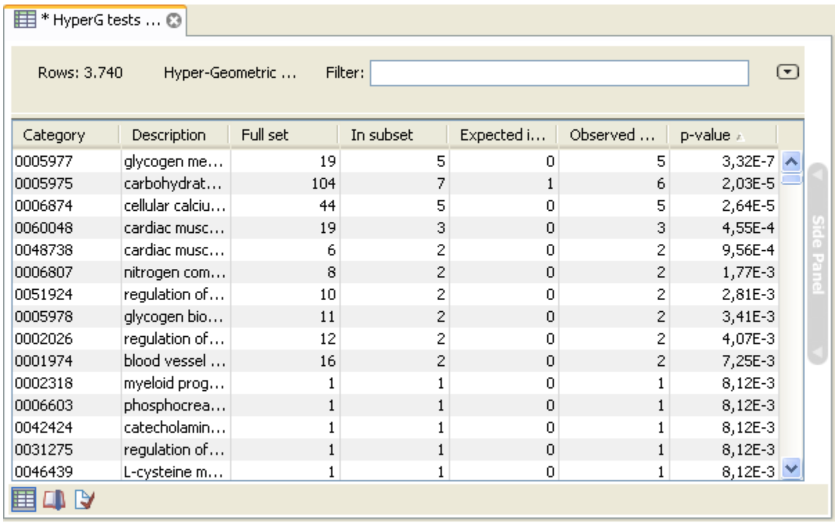
○ Original expression values

● Transformed expression values

○ Normalized expression values

Figure 2.125: *Testing on GO biological process.*

Click **Next** and **Finish** to perform the test. The result is shown in figure 2.126.



* HyperG tests ...

Rows: 3.740 Hyper-Geometric ... Filter:

Category	Description	Full set	In subset	Expected i...	Observed ...	p-value
0005977	glycogen me...	19	5	0	5	3,32E-7
0005975	carbohydrat...	104	7	1	6	2,03E-5
0006874	cellular calciu...	44	5	0	5	2,64E-5
0060048	cardiac musc...	19	3	0	3	4,55E-4
0048738	cardiac musc...	6	2	0	2	9,56E-4
0006807	nitrogen com...	8	2	0	2	1,77E-3
0051924	regulation of...	10	2	0	2	2,81E-3
0005978	glycogen bio...	11	2	0	2	3,41E-3
0002026	regulation of...	12	2	0	2	4,07E-3
0001974	blood vessel ...	16	2	0	2	7,25E-3
0002318	myeloid prog...	1	1	0	1	8,12E-3
0006603	phosphocrea...	1	1	0	1	8,12E-3
0042424	catecholamin...	1	1	0	1	8,12E-3
0031275	regulation of...	1	1	0	1	8,12E-3
0046439	L-cysteine m...	1	1	0	1	8,12E-3

Figure 2.126: *The result of testing on GO biological process.*

This table lists the GO categories according to p-values for this test. If you take number 2, carbohydrate metabolic process, there are 104 genes in this category in the full set, if the subset was randomly chosen you would have expected 1 gene to be in the subset. But because there are 7 genes in this subset, this process is over-represented and given a p-value of 2.63E-5.

2.18.4 A different approach: Gene Set Enrichment Analysis (GSEA)

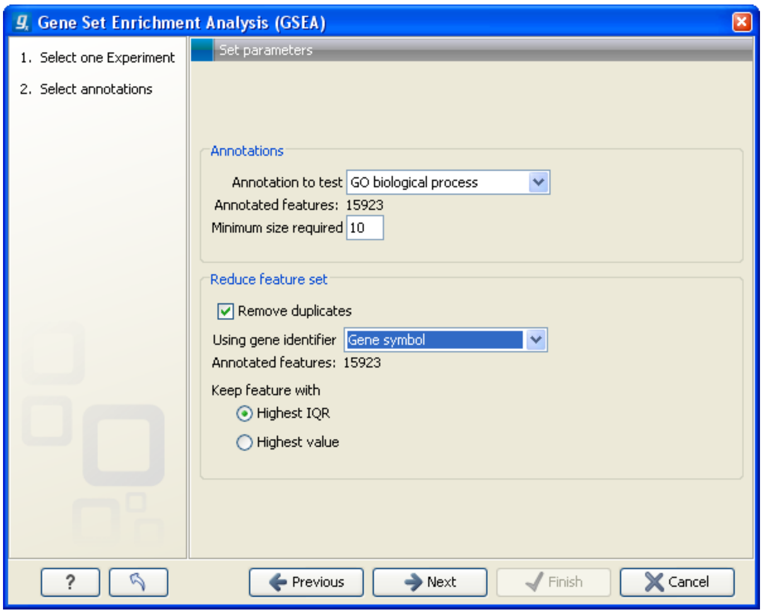
The hypergeometric tests on annotations uses a pre-defined subset of differentially expressed genes as a starting point and compares the annotations in this list to those of the genes in the full experiment. The exact limit for this subset is somewhat arbitrary - in our case we could have chosen a p-value less than 0.005 instead of 0.0005 and it would lead to a different result.

Furthermore, only the most apparently differentially expressed genes are used in the subset - one could easily imagine that other categories would be significant based on more genes with e.g. lower fold change or higher p-values.

The Gene Set Enrichment Analysis (GSEA) does not take an *a priori* defined list of differentially expressed genes and compares it to the full list - it uses a single experiment. It ranks the genes on p-value and analyzes whether there are some categories that are over-represented in the top of the list.

> **Toolbox | Expression Analysis ( ) | Annotation Test | Gene Set Enrichment Analysis (GSEA) ( )**

Select the original full experiment and click **Next**. In this step, make sure the **GO biological process** is chosen (see figure 2.127.



Annotations

Annotation to test: GO biological process

Annotated features: 15923

Minimum size required: 10

Reduce feature set

☑ Remove duplicates

Using gene identifier: Gene symbol

Annotated features: 15923

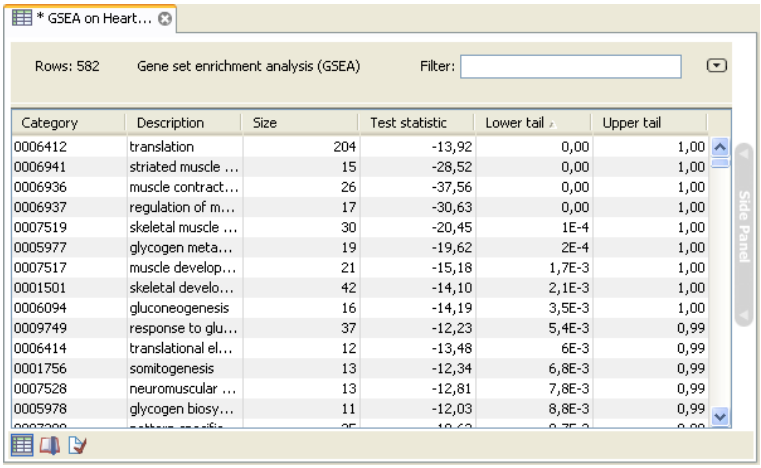
Keep feature with

● Highest IQR

○ Highest value

Figure 2.127: *Gene set enrichment analsysis based on GO biological process.*

Click **Next** and select the **Transformed expression values**. Click **Finish**. The result is shown in figure 2.128.



Rows: 582   Gene set enrichment analysis (GSEA)   Filter:

Category	Description	Size	Test statistic	Lower tail	Upper tail
0006412	translation	204	-13,92	0,00	1,00
0006941	striated muscle ...	15	-28,52	0,00	1,00
0006936	muscle contract...	26	-37,56	0,00	1,00
0006937	regulation of m...	17	-30,63	0,00	1,00
0007519	skeletal muscle ...	30	-20,45	1E-4	1,00
0005977	glycogen meta...	19	-19,62	2E-4	1,00
0007517	muscle develop...	21	-15,18	1,7E-3	1,00
0001501	skeletal develo...	42	-14,10	2,1E-3	1,00
0006094	gluconeogenesis	16	-14,19	3,5E-3	1,00
0009749	response to glu...	37	-12,23	5,4E-3	0,99
0006414	translational el...	12	-13,48	6E-3	0,99
0001756	somitogenesis	13	-12,34	6,8E-3	0,99
0007528	neuromuscular ...	13	-12,81	7,8E-3	0,99
0005978	glycogen biosy...	11	-12,03	8,8E-3	0,99

Figure 2.128: *The result of a gene set enrichment analsysis based on GO biological process.*

The table is sorted on the lower tail so that the GO categories where up-regulated genes in the first group are over-represented are placed at the top, and the GO categories where up-regulated genes in the second group are over-represented are placed at the bottom.

Note that we could have chosen to filter away genes with less reliable measurements from the experiment (as shown in the previous tutorial) before subjecting it to the GSEA analysis in order to limit noise and aim for a more robust result.

2.19 Tutorial: Assembly

In this tutorial, you will see how to assemble data from automated sequencers into a contig and how to find and inspect any conflicts that may exist between different reads.

This tutorial shows how to assemble sequencing data generated by conventional "Sanger" sequencing techniques. For high-throughput sequencing data, we refer to the *CLC Genomics Workbench* (see `http://www.clcbio.com/genomics`).

The data used in this tutorial are the sequence reads in the "Sequencing reads (raw data)" folder in the "Sequencing data" folder of the **Example data**.

2.19.1 Trimming the sequences

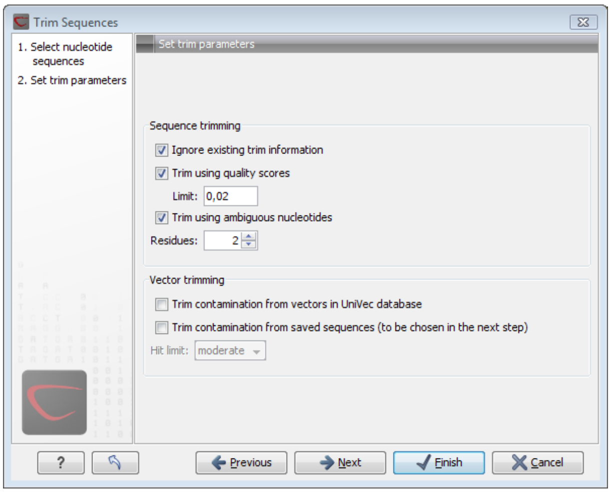
The first thing to do when analyzing sequencing data is to trim the sequences. Trimming serves a dual purpose: it both takes care of parts of the reads with poor quality, and it removes potential vector contamination. Trimming the sequencing data gives a better result in the further analysis.

Toolbox in the Menu Bar | Sequencing Data Analyses ( ) | Trim Sequences ( )

Select the 9 sequences and click **Next**.

In this dialog, you will be able to specify how this trimming should be performed.

For this data, we wish to use a more stringent trimming, so we set the limit of the quality score trim to 0,02 (see figure 2.129).



Trim Sequences

1. Select nucleotide sequences
2. Set trim parameters

Set trim parameters

Sequence trimming

☑ Ignore existing trim information

☑ Trim using quality scores

Limit: 0,02

☑ Trim using ambiguous nucleotides

Residues: 2

Vector trimming

□ Trim contamination from vectors in UniVec database

□ Trim contamination from saved sequences (to be chosen in the next step)

Hit limit: moderate

Figure 2.129: *Specifying how sequences should be trimmed. A stringent trimming of 0,02 is used in this example.*

There is no vector contamination in these data, se we only trim for poor quality.

If you place the mouse cursor on the parameters, you will see a brief explanation.

Click **Next** and choose to **Save** the results. When the trimming is performed, the parts of the sequences that are trimmed away will be annotated with an annotation (see figure 2.130). By choosing **Save**, the annotations will be saved directly to the sequences without opening them.

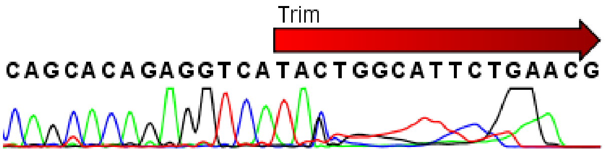


Figure 2.130: *Trimming creates annotations on the regions that will be ignored in the assembly process.*

These annotated parts of the sequences will be ignored in the subsequent assembly.

A natural question that comes to mind at this point is: Why not simply delete the trimmed regions instead of annotating them? In some cases, deleting the regions would do no harm, but in other cases, these regions could potentially contain valuable information, and this information would be lost if the regions were deleted instead of annotated. We will see an example of this later in this tutorial.

2.19.2 Assembling the sequencing data

The next step is to assemble the sequences. This is the technical term for aligning the sequences where they overlap and reverse the reverse reads to make a contiguous sequence (also called a contig).

In this tutorial, we will use assembly to a reference sequence. This can be used when you have a reference sequence that you know is similar to your sequencing data.

> **Toolbox in the Menu Bar | Sequencing Data Analyses ( ) | Assemble Sequences to Reference ( )**

In the first dialog, select the nine sequencing reads and click **Next** to go to the second step of the assembly where you select the reference sequence.

Click the **Browse and select** button ( ) and select the "ATP8a1 mRNA (reference)" from the "Sequencing data" folder (see figure 2.131).

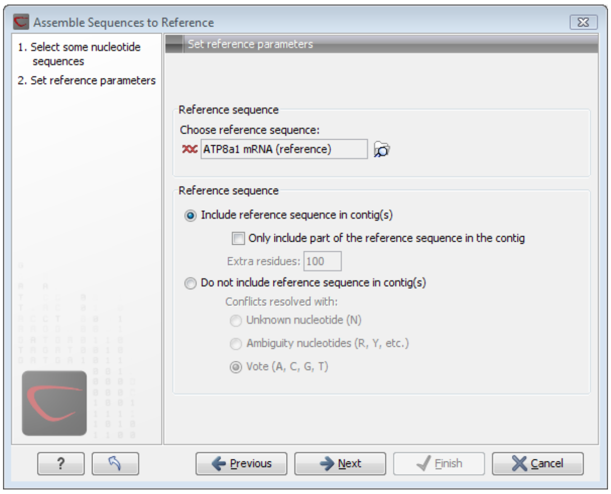
Click **Next** and choose to use the trim information (that you have just added). Click **Finish** and the assembly process begins.

2.19.3 Getting an overview of the contig

The result of the assembly is a **Contig** which is an alignment of the nine reads to the reference sequence. Click **Fit width** ( ) to see an overview of the contig. To help you determine the coverage, display a coverage graph (see figure 2.132):

> **Alignment info in Side Panel | Coverage | Graph**

This overview can be an aid in determining whether coverage is satisfactory, and if not, which



Assemble Sequences to Reference

1. Select some nucleotide sequences
2. Set reference parameters

Set reference parameters

Reference sequence

Choose reference sequence:

ATP8a1 mRNA (reference)

Reference sequence

● Include reference sequence in contig(s)

□ Only include part of the reference sequence in the contig

Extra residues: 100

○ Do not include reference sequence in contig(s)

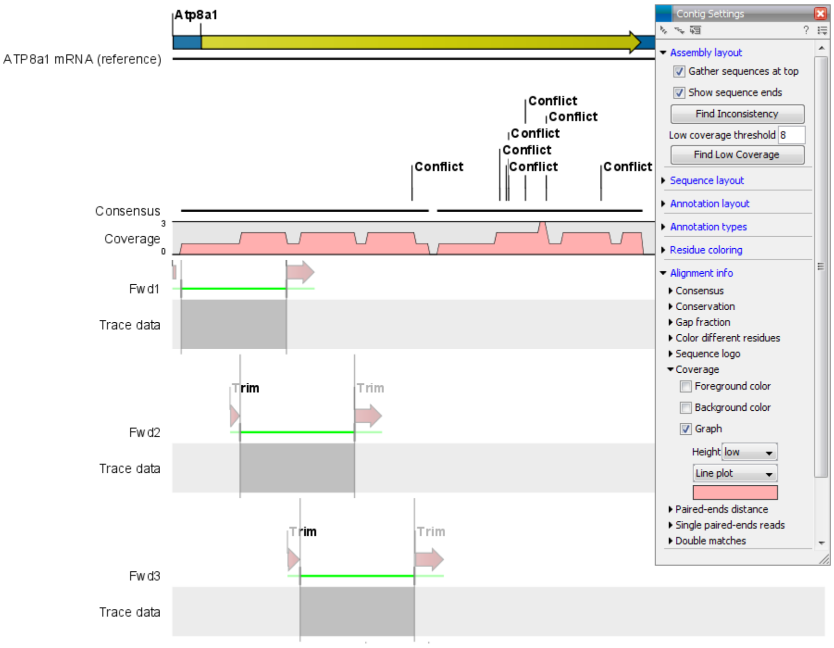
Conflicts resolved with:

○ Unknown nucleotide (N)

○ Ambiguity nucleotides (R, Y, etc.)

● Vote (A, C, G, T)

Figure 2.131: *The "ATP8a1 mRNA (reference)" sequence selected as reference sequence for the assembly.*



Contig Settings

▾ Assembly layout

☑ Gather sequences at top

☑ Show sequence ends

Find Inconsistency

Low coverage threshold 8

Find Low Coverage

▸ Sequence layout

▸ Annotation layout

▸ Annotation types

▸ Residue coloring

▾ Alignment info

▸ Consensus

▸ Conservation

▸ Gap fraction

▸ Color different residues

▸ Sequence logo

▾ Coverage

□ Foreground color

□ Background color

☑ Graph

Height low

Line plot

▸ Paired-ends distance

▸ Single paired-ends reads

▸ Double matches

Figure 2.132: *An overview of the contig with the coverage graph.*

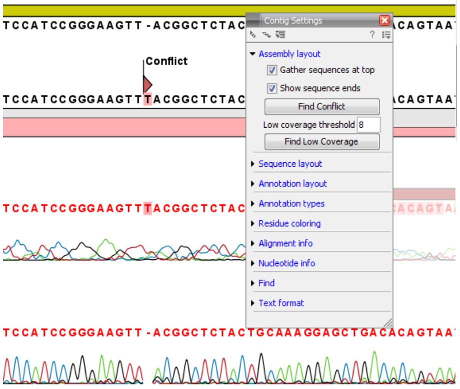
regions a new sequencing effort should focus on. Next, we go into the details of the contig.

2.19.4 Finding and editing conflicts

Click **Zoom to 100%** ( ) to zoom in on the residues at the beginning of the contig. Click the **Find Conflict** button at the top of the **Side Panel** or press the **Space** key to find the first position where there is disagreement between the reads (see figure 2.133).

In this example, the first read has a "T" (marked with a light-pink background color), whereas the second line has a gap. In order to determine which of the reads we should trust, we assess the quality of the read at this position.

A quick look at the regularity of the peaks of read "Rev2" compared to "Rev3" indicates that we



Contig Settings

▾ Assembly layout

☑ Gather sequences at top

☑ Show sequence ends

Find Conflict

Low coverage threshold 8

Find Low Coverage

▸ Sequence layout

▸ Annotation layout

▸ Annotation types

▸ Residue coloring

▸ Alignment info

▸ Nucleotide info

▸ Find

▸ Text format

TCCATCCGGGAAGTT-ACGGCTCTAC
CAGTAA
Conflict
TCCATCCGGGAAGTTTACGGCTCTAC
CAGTAA
TCCATCCGGGAAGTTTACGGCTCTAC
ACAGTA
TCCATCCGGGAAGTT-ACGGCTCTACTGCAAAGGAGCTGACACAGTAA

Figure 2.133: *Using the Find Conflict button highlights conflicts.*

should trust the "Rev2" read. In addition, you can see that we are close to the end of the end of "Rev3", and the quality of the chromatogram traces is often low near the ends.

Based on this, we decide not to trust "Rev3". To correct the read, select the "T" in the "Rev3" sequence and press **Delete** (🗋).

This will resolve the conflict.

Clicking the **Find Conflict** button again will find the next conflict.

2.19.5 Including regions that have been trimmed off

This is the beginning of a stretch of gaps in the consensus sequence. This is because the reads have been trimmed at this position. However, if you look at the read at the bottom, *Fwd2*, you can see that a lot of the peaks seem actually to be fine, so we could just as well include this information in the contig.

If you scroll a little to the right, you can see where the trimmed-off region begins. To re-include this region in the contig, move the vertical slider at position 2073 to the left (see figure 18.11).

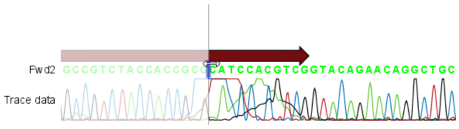


Figure 2.134: *Dragging the edge of the trimmed region.*

You will now see how the gaps in the consensus sequence are replaced by real sequence information.

Note that you can only move the sliders when you are zoomed in to see the sequence residues.

Clicking the **Find Conflict** button again will find the next conflict.

2.19.6 Inspecting the traces

Here both reads are different than the reference sequence. We now inspect the traces in more detail. In order to see the details, we zoom in on this position:

Zoom in in the Tool Bar (🔍) | Click the selected base | Click again three times

Now you have zoomed in on the trace (see figure 2.135).

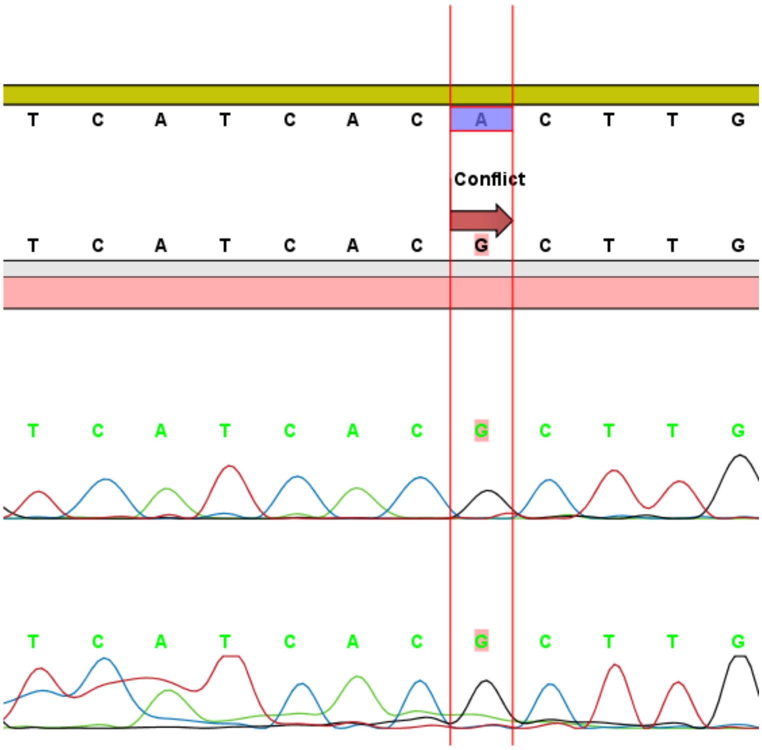


Figure 2.135: *Now you can see all the details of the traces.*

This gives more space between the residues, but if we would like to inspect the peaks even more, simply drag the peaks up and down with your mouse (see figure 18.2).

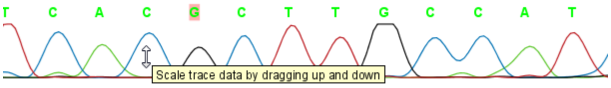

T C A C G C T T G C C A T
Scale trace data by dragging up and down

Figure 2.136: *Grab the traces to scale.*

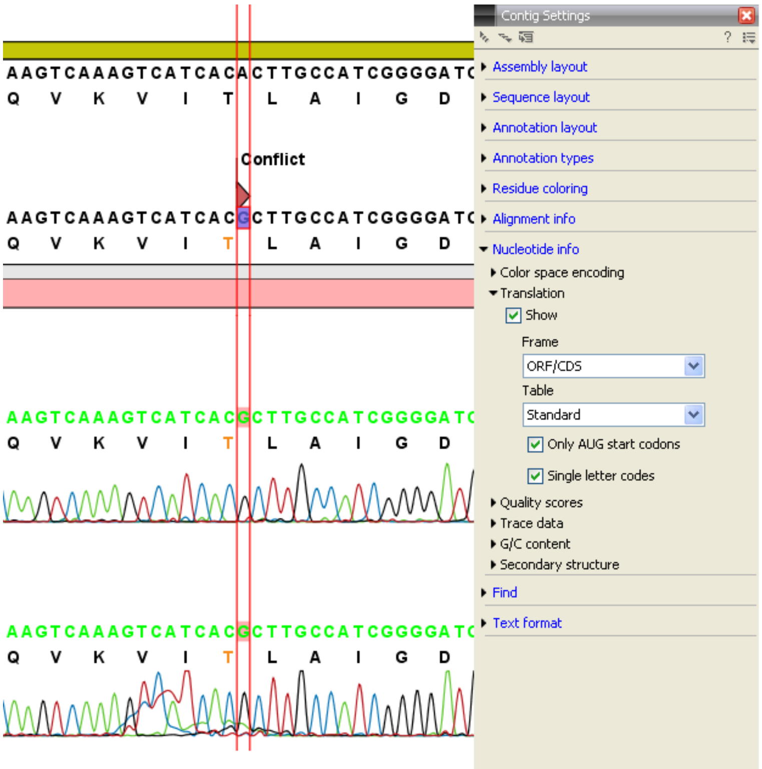
2.19.7 Synonymous substitutions?

In this case we have sequenced the coding part of a gene. Often you want to know if a variation like this would mean on the protein level. To do this, show the translation along the contig:

> **Nucleotide info in the Side Panel | Translation | Show | Select ORF/CDS in the Frame box**

The result is shown in figure 2.137.

You can see that the variation is on the third base of the codon coding for Threonine, so this is a synonymous substitution. That is why the T is colored yellow. If it was a non-synonymous substitution, it would be colored in red.



☑ Show

Frame: ORF/CDS

Table: Standard

☑ Only AUG start codons

☑ Single letter codes

Figure 2.137: *Showing the translation along the contig.*

2.19.8 Getting an overview of the conflicts

Browsing the conflicts by clicking the **Find Conflict** button is useful in many cases, but you might also want to get an overview of all the conflicts in the entire contig. This is easily achieved by showing the contig in a table view:

> **Press and hold the Ctrl-button (⌘ on Mac) | Click Show Table () at the bottom of the view**

This will open a table showing the conflicts. You can right-click the **Note** field and enter your own comment. In this dialog, enter a new text in the **Name** and click **OK**.

When you edit a comment, this is reflected in the conflict annotation on the consensus sequence. This means that when you use this sequence later on, you will easily be able to see the comments you have entered. The comment could be e.g. your interpretation of the conflict.

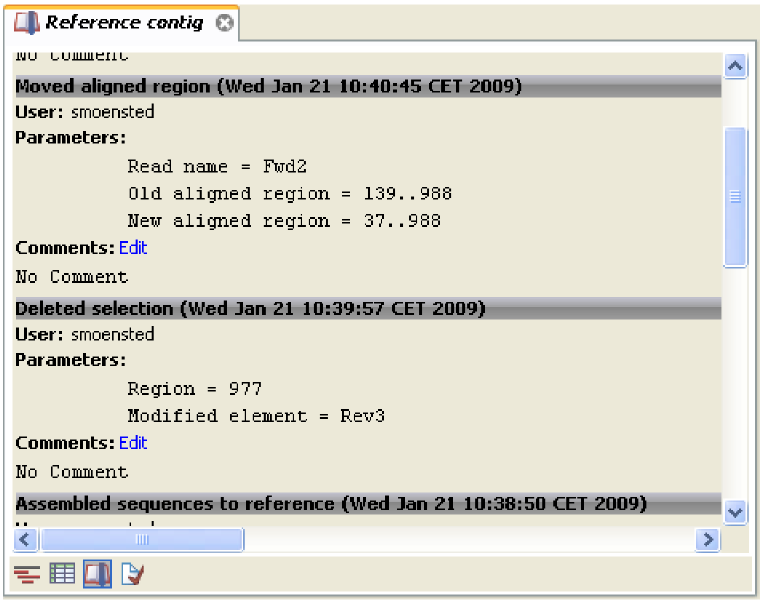
2.19.9 Documenting your changes

Whenever you make a change like deleting a "T", it will be noted in the contig's history. To open the history, click the fHistory () icon at the bottom of the view.

In the history, you can see the details of each change (see figure 2.138).

2.19.10 Using the result for further analyses

When you have finished editing the contig, it can be saved, and you can also extract and save the consensus sequence:



Reference contig

```
No Comment
```

Moved aligned region (Wed Jan 21 10:40:45 CET 2009)

User: smoensted

Parameters:

```
          Read name = Fwd2
          Old aligned region = 139..988
          New aligned region = 37..988
```

Comments: Edit

```
No Comment
```

Deleted selection (Wed Jan 21 10:39:57 CET 2009)

User: smoensted

Parameters:

```
          Region = 977
          Modified element = Rev3
```

Comments: Edit

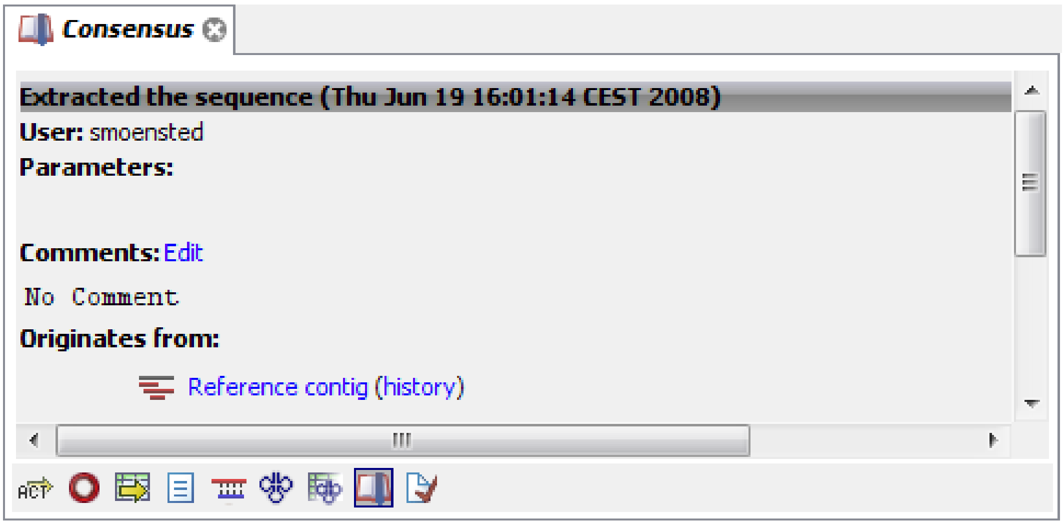
```
No Comment
```

Assembled sequences to reference (Wed Jan 21 10:38:50 CET 2009)

Figure 2.138: *The history of the contig showing that a "T" has been deleted and that the aligned region has been moved.*

Right-click the name "Consensus" | Open Copy of Sequence | Save ( )

This will make it possible to use this sequence for further analyses in the *CLC Genomics Workbench*. All the conflict annotations are preserved, and in the sequence's history, you will find a reference to the original contig. As long as you also save the contig, you will always be able to go back to it by clicking the reference in the consensus sequence's history (see figure 2.139).



Consensus

Extracted the sequence (Thu Jun 19 16:01:14 CEST 2008)

User: smoensted

Parameters:

Comments: Edit

```
No Comment
```

Originates from:

Reference contig (history)

Figure 2.139: *The history of the consensus sequence which has been extracted from the contig. Clicking the blue text will find the saved contig in the Navigation Area.*

2.20 Tutorial: In silico cloning cloning work flow

In this tutorial, the goal is to PCR-amplify a gene using primers with restriction sites at the 5' end, and insert the gene into a multiple cloning site of an expression vector. We start off with a set of primers, a DNA template sequence and an expression vector, and this tutorial will guide you through the following steps:

1. Add restriction sites to the primers
2. Simulate a PCR to create the fragment to use for cloning
3. Specify restriction sites to use for cloning and insert the fragment into the vector

2.20.1 Locating the data to use

Open the `Example data` folder in the **Navigation Area** (if you do not have the example data, please go to the **Help** menu to install it). Open the `Cloning` folder, and inside this folder, open the `Primer` folder.

The data to use in these folders is shown in figure 2.140.

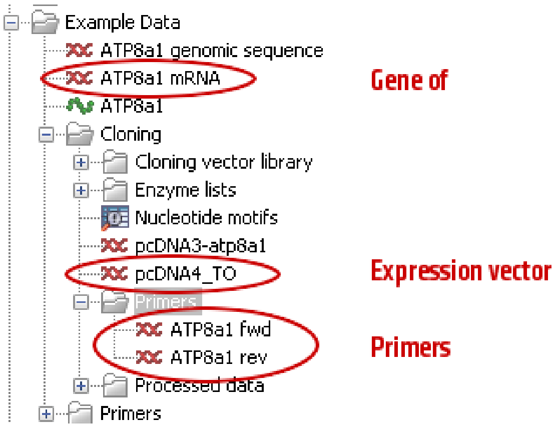


Figure 2.140: *The data to use in this tutorial.*

Double-click the `ATP8a1 mRNA` sequence and zoom to **Fit Width** ( ) and you will see the yellow annotation which is the coding part of the gene. This is the part that we want to insert into the `pcDNA4_TO` vector. The primers have already been designed using the primer design tool in *CLC Genomics Workbench* (to learn more about this, please check the Primer design tutorial).

2.20.2 Add restriction sites to primers

First, we add restriction sites to the primers. In order to see which restriction enzymes can be used, we create a split view of the vector and the fragment to insert. In this way we can easily make a visual check to find enzymes from the multiple cloning site in the vector that do not cut in the gene of interest. To create the split view:

double-click the pcDNA4_TO sequence | **View** | **Split Horizontally** (**)**

Note that this can also be achieved by simply dragging the `pcDNA4_TO` sequence into the lower part of the open view.

Switch to the **Circular** (⭘) view at the bottom of the view.

Zoom in (🔍) on the multiple cloning site downstream of the green `CMV` promoter annotation. You should now have a view similar to the one shown in figure 2.141.

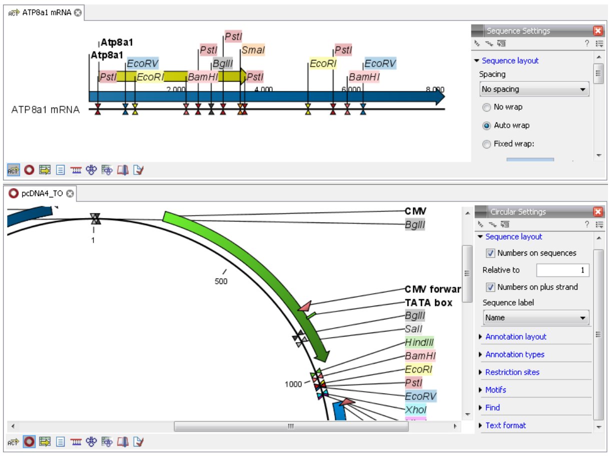


Figure 2.141: *Check cut sites.*

By looking at the enzymes we can see that both *HindIII* and *XhoI* cut in the multi-cloning site of the vector and not in the Atp8a1 gene. Note that you can add more enzymes to the list in the **Side Panel** by clicking **Manage Enzymes** under the **Restriction Sites** group.

Close both views and open the `ATP8a1 fwd` primer sequence. When it opens, double-click the name of the sequence to make a selection of the full sequence. Then right-click and **Add Restriction Site Before Selection** as shown in figure 2.142.

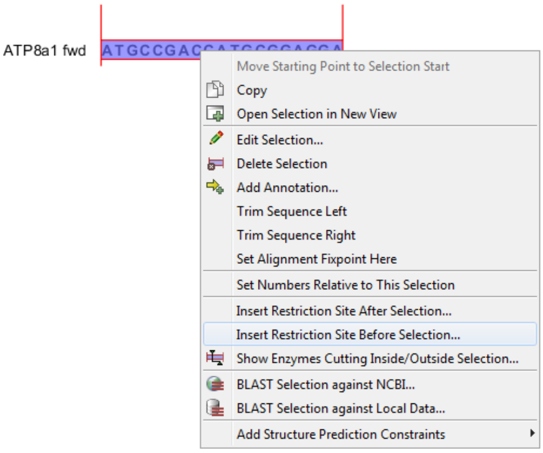
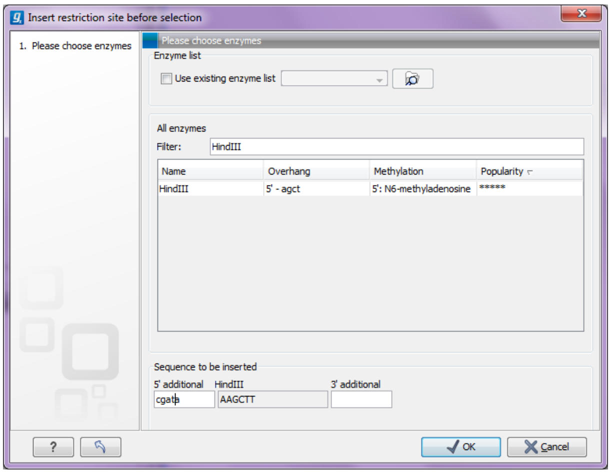

ATP8a1 fwd
ATGCCGACCATCCCACCA
Move Starting Point to Selection Start
Copy
Open Selection in New View
Edit Selection...
Delete Selection
Add Annotation...
Trim Sequence Left
Trim Sequence Right
Set Alignment Fixpoint Here
Set Numbers Relative to This Selection
Insert Restriction Site After Selection...
Insert Restriction Site Before Selection...
Show Enzymes Cutting Inside/Outside Selection...
BLAST Selection against NCBI...
BLAST Selection against Local Data...
Add Structure Prediction Constraints

Figure 2.142: *Adding restriction sites to a primer.*

In the **Filter** box enter *HindIII* and select it. At the bottom of the dialog, add a few extra bases 5' of the cut site (this is done to increase the efficiency of the enzyme) as shown figure 21.14.

Click **OK** and the sequence will be inserted at the 5' end of the primer as shown in figure figure 2.144.

Perform the same process for the `ATP8a1 rev` primer, this time using *XhoI* instead. This time, you should also add a few bases at the 5' end as was done in figure21.14 when inserting the *HindIII* site.



Insert restriction site before selection

1. Please choose enzymes

Please choose enzymes

Enzyme list

□ Use existing enzyme list

All enzymes

Filter: HindIII

Name	Overhang	Methylation	Popularity
HindIII	5' - agct	5': N6-methyladenosine	*****

Sequence to be inserted

5' additional	HindIII	3' additional
cgata	AAGCTT	

Figure 2.143: *Adding restriction sites to a primer.*

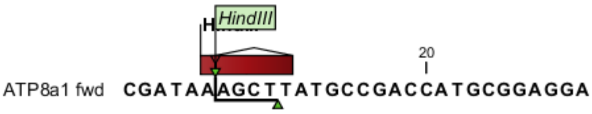


Figure 2.144: *Adding restriction sites to a primer.*

Note! The `ATP8a1 rev` primer is designed to match the negative strand, so the restriction site should be added at the 5' end of this sequence as well (**Add Restriction Site before Selection**).

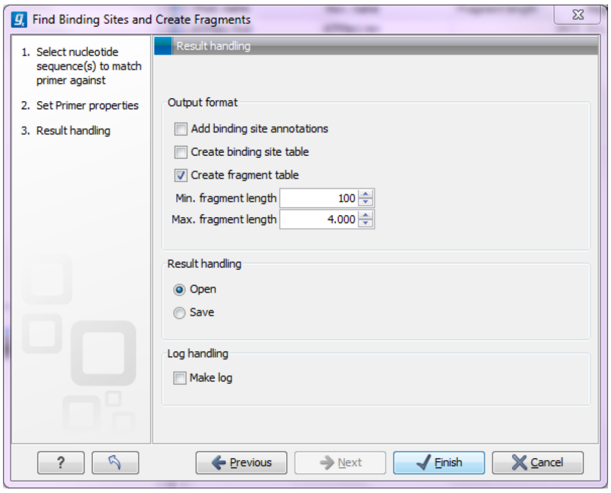
Save (💾) the two primers and close the views and you are ready for next step.

2.20.3 Simulate PCR to create the fragment

Now, we want to extract the PCR product from the template `ATP8a1 mRNA` sequence using the two primers with restriction sites:

Toolbox | **Primers and Probes** ( ) | **Find Binding Sites and Create Fragments** ( )

Select the `ATP8a1 mRNA` sequence and click **Next**. In this dialog, use the **Browse** ( ) button to select the two primer sequences. Click **Next** and adjust the output options as shown in figure 2.145.



Find Binding Sites and Create Fragments

1. Select nucleotide sequence(s) to match primer against
2. Set Primer properties
3. Result handling

Result handling

Output format

□ Add binding site annotations

□ Create binding site table

☑ Create fragment table

Min. fragment length 100

Max. fragment length 4.000

Result handling

● Open

○ Save

Log handling

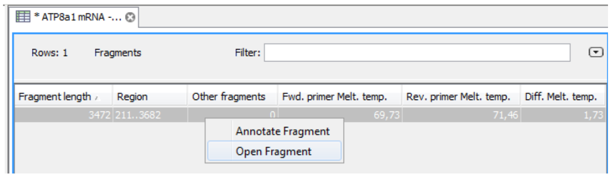
□ Make log

Figure 2.145: *Creating the fragment table including fragments up to 4000 bp.*

Click **Finish** and you will now see the fragment table displaying the PCR product.

In the **Side Panel** you can choose to show information about melting temperature for the primers.

Right-click the fragment and select **Open Fragment** as shown in figure 2.146.



Rows: 1 Fragments Filter:

Fragment length	Region	Other fragments	Fwd. primer Melt. temp.	Rev. primer Melt. temp.	Diff. Melt. temp.
3472	211..3682	0	69,73	71,46	1,73

Figure 2.146: *Opening the fragment as a sequence.*

This will create a new sequence representing the PCR product. **Save** (💾) the sequence in the `Cloning` folder and close the views. You do not need to save the fragment table.

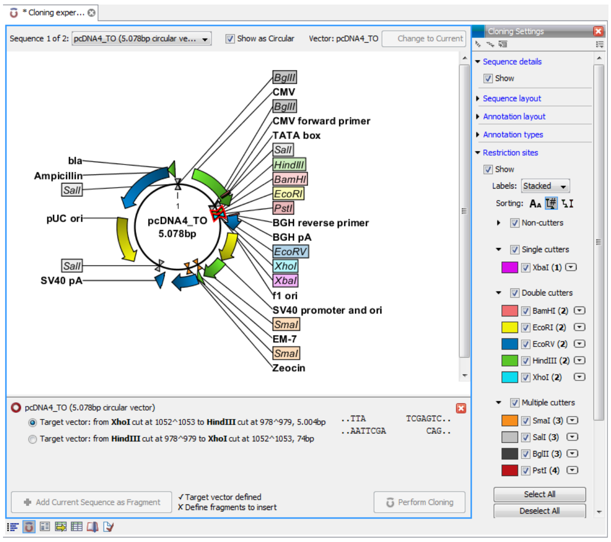
2.20.4 Specify restriction sites and perform cloning

The final step in this tutorial is to insert the fragment into the cloning vector:

Toolbox in the Menu Bar | Cloning and Restriction Sites ( ) | Cloning ( )

Select the `Fragment (ATP8a1 mRNA (ATP8a1 fwd - ATP8a1 rev))` sequence you just saved and click **Next**. In this dialog, use the **Browse** ( ) button to select `pcDNA4_TO` cloning vector also located in the `Cloning` folder. Click **Finish**.

You will now see the cloning editor where you will see the `pcDNA4_TO` vector in a circular view. Press and hold the Ctrl (⌘ on Mac) key while you click first the *HindIII* site and next the *XhoI* site (see figure 2.147).



pcDNA4_TO (5.078bp circular vector)

● Target vector: from **XhoI** cut at 1052^1053 to **HindIII** cut at 978^979, 5.004bp

○ Target vector: from **HindIII** cut at 978^979 to **XhoI** cut at 1052^1053, 74bp

```
..TTA        TCGAGTC..
..AATTCGA         CAG..
```

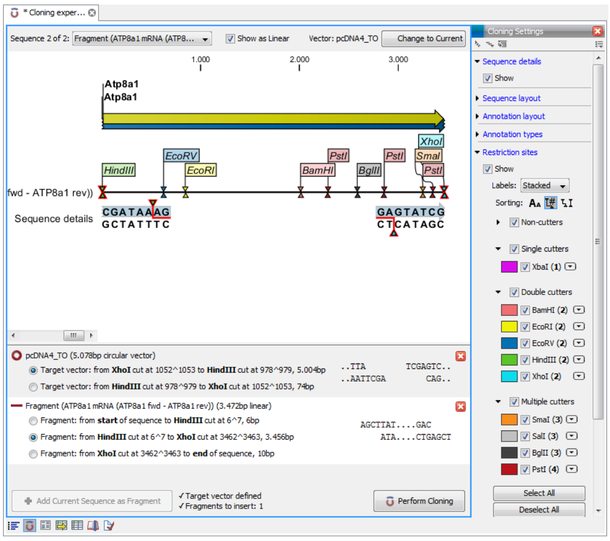
✓ Target vector defined
X Define fragments to insert

Sequence details
☑ Show
Sequence layout
Annotation layout
Annotation types
Restriction sites
☑ Show
Labels: Stacked
Sorting:
☑ Non-cutters
☑ Single cutters
☑ XbaI (1)
☑ Double cutters
☑ BamHI (2)
☑ EcoRI (2)
☑ EcoRV (2)
☑ HindIII (2)
☑ XhoI (2)
☑ Multiple cutters
☑ SmaI (3)
☑ SalI (3)
☑ BglII (3)
☑ PstI (4)

Figure 2.147: *Press and hold the Ctrl key while you click first the* HindIII *site and next the* XhoI *site.*

At the bottom of the view you can now see information about how the vector will be cut open. Since the vector has now been split into two fragments, you can decide which one to use as the target vector. If you selected first the *HindIII* site and next the *XhoI* site, the *CLC Genomics Workbench* has already selected the right fragment as the target vector. If you click one of the vector fragments, the corresponding part of the sequence will be high-lighted.

Next step is to cut the fragment. At the top of the view you can switch between the sequences used for cloning (at this point it says `pcDNA4_TO 5.078bp circular vector`). Switch to the fragment sequence and perform the same selection of cut sites as before while pressing the Ctrl (⌘ on Mac) key. You should now see a view identical to the one shown in figure 2.148.



pcDNA4_TO (5.078bp circular vector)

- ● Target vector: from **XhoI** cut at 1052^1053 to **HindIII** cut at 978^979, 5.004bp ..TTA ..AATTCGA TCGAGTC.. CAG..
- ○ Target vector: from **HindIII** cut at 978^979 to **XhoI** cut at 1052^1053, 74bp

Fragment (ATP8a1 mRNA (ATP8a1 fwd - ATP8a1 rev)) (3.472bp linear)

- ○ Fragment: from **start** of sequence to **HindIII** cut at 6^7, 6bp
- ● Fragment: from **HindIII** cut at 6^7 to **XhoI** cut at 3462^3463, 3.456bp AGCTTAT....GAC ATA....CTGAGCT
- ○ Fragment: from **XhoI** cut at 3462^3463 to **end** of sequence, 10bp

✓Target vector defined
✓Fragments to insert: 1

Figure 2.148: *Press and hold the Ctrl key while you click first the* HindIII *site and next the* XhoI *site.*

When this is done, the **Perform Cloning** ( ) button at the lower right corner of the view is active because there is now a valid selection of both fragment and target vector. Click the **Perform Cloning** ( ) button and you will see the dialog shown in figure 2.149.

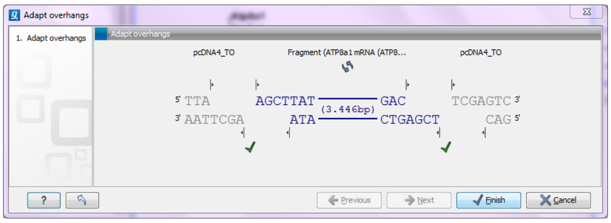


Figure 2.149: *Showing the insertion point of the vector*

This dialog lets you inspect the overhangs of the cut site, showing the vector sequence on each side and the fragment in the middle. The fragment can be reverse complemented by clicking the **Reverse complement fragment** ( ) but this is not necessary in this case. Click **Finish** and your new construct will be opened.

When saving your work, there are two options:

- Saving the sequence list used to specify cut sites is useful if you need to perform the same process again or double-check details.
- Saving the construct shown in the circular view will only save the information on the particular sequence including details about how it was created (this can be shown in the **History** view).

You can of course also choose to save both. In that case, the history of the construct will point to the sequence list in its own history.

The construct is shown in figure 2.150.

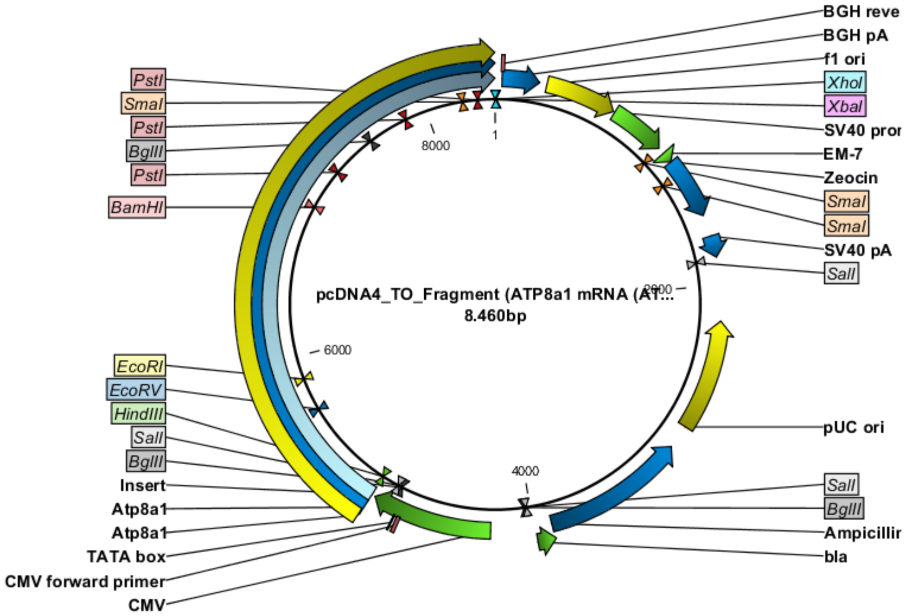


Figure 2.150: *The Atp8a1 gene inserted after the CMV promoter*

2.21 Tutorial: Primer design

In this tutorial, you will see how to use *CLC Genomics Workbench* for finding primers for PCR amplification of a specific region.

We use the pcDNA3-atp8a1 sequence from the 'Primers' folder in the Example data. This sequence is the pcDNA3 vector with the atp8a1 gene inserted. In this tutorial, we wish to get a PCR product covering the insertion point of the gene. The reason for doing this is that we want to use the PCR to check that the gene is actually inserted where we think it is.

First, open the sequence in the Primer Designer:

> **Select the pcDNA3-atp8a1 sequence | Show ( ) | Primer Designer ( )**

Now the sequence is opened and we are ready to begin designing primers.

2.21.1 Specifying a region for the forward primer

First zoom out to get an overview of the sequence by clicking **Fit Width** ( ). You can now see the blue gene annotation labeled Atp8a1, and just before that there is the green CMV promoter.

In this tutorial, we want the forward primer to be in a region between positions 600 and 900 - just before the gene (you may have to zoom in ( ) to make the selection). Select this region, right-click and choose "Forward primer region here" ( ) (see figure 2.151).

This will add an annotation to this region, and five rows of red and green dots are seen below as shown in figure 2.152:

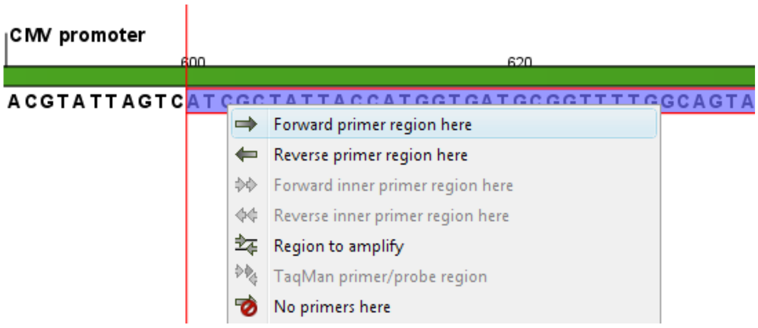

CMV promoter
600
620
ACGTATTAGTCATCGCTATTACCATGGTGATGCGGTTTTGGCAGTA
Forward primer region here
Reverse primer region here
Forward inner primer region here
Reverse inner primer region here
Region to amplify
TaqMan primer/probe region
No primers here

Figure 2.151: *Right-clicking a selection and choosing "Forward primer region here".*

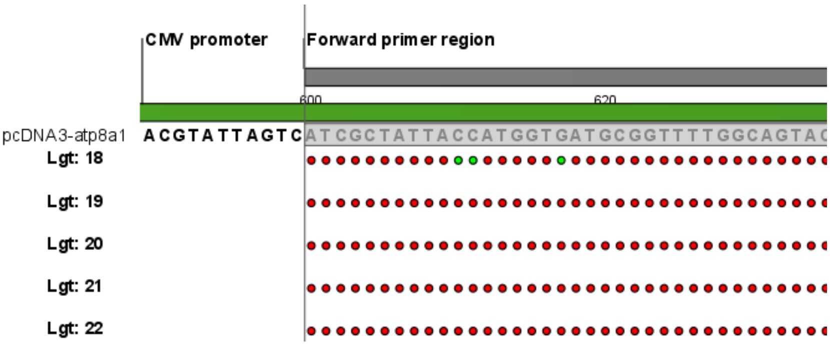


Figure 2.152: *Five lines of dots representing primer suggestions. There is a line for each length.*

2.21.2 Examining the primer suggestions

Each line consists of a number of dots, each representing the *starting point* of a possible primer. E.g. the first dot on the first line (primers of length 18) represents a primer starting at the dot's position and with a length of 18 nucleotides (shown as the white area in figure 2.153):

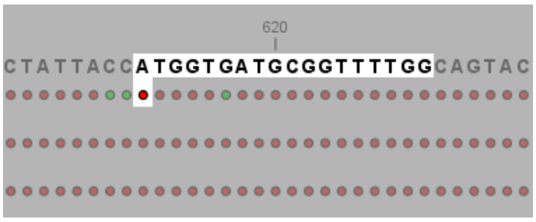


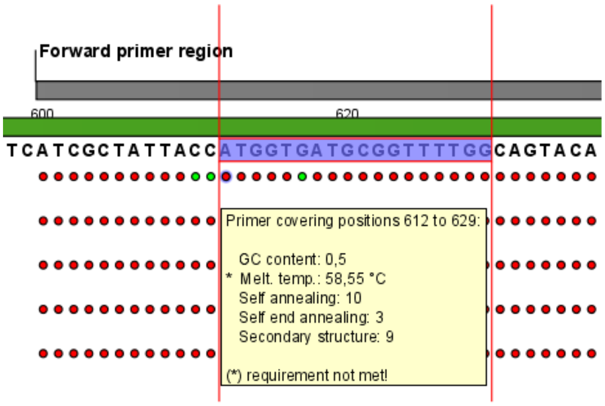
Figure 2.153: *The first dot on line one represents the starting point of a primer that will anneal to the highlighted region.*

Position the mouse cursor upon a dot and you will see an information box providing data about this primer. Clicking the dot will select the region where the primer will anneal. (See figure 2.154):

Note that some of the dots are colored red. This indicates that the primer represented by this dot does not meet the requirements set in the **Primer parameters** (see figure 2.155):

Note that the maximum melting temperature is per default set to 58, and this is the reason why the primer in figure 2.154 with a melting temperature of 58,55 does not meet the requirements and is colored red. If you raise the maximum melting temperature to 59, the primer will meet the requirements and the dot becomes green.

In figure 2.154 there is an asterisk (*) before the melting temperature. This indicates that this primer does not meet the requirements regarding melting temperature. In this way, you can easily

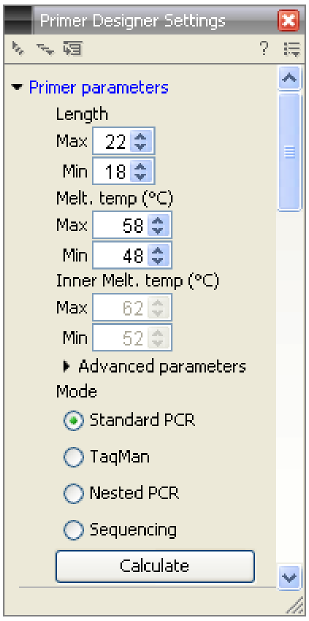


Primer covering positions 612 to 629:

GC content: 0,5
* Melt. temp.: 58,55 °C
Self annealing: 10
Self end annealing: 3
Secondary structure: 9

(*) requirement not met!

Figure 2.154: *Clicking the dot will select the corresponding region, and placing the cursor upon the dot will reveal an information box.*



Length
Max 22
Min 18
Melt. temp (°C)
Max 58
Min 48
Inner Melt. temp (°C)
Max 62
Min 52

Mode
● Standard PCR
○ TaqMan
○ Nested PCR
○ Sequencing

Figure 2.155: *The Primer parameters.*

see why a specific primer (represented by a dot) fails to meet the requirements.

By adjusting the **Primer parameters** you can define primers which match your specific needs. Since the dots are constantly updated, you can immediately see how a change in the primer parameters affects the number of red and green dots.

2.21.3 Calculating a primer pair

Until now, we have been looking at the forward primer. To mark a region for the reverse primer, make a selection from position 1200 to 1400 and:

Right-click the selection | Reverse primer region here (⇐)

The two regions should now be located as shown in figure 2.156:

Now, you can let *CLC Genomics Workbench* calculate all the possible primer pairs based on the **Primer parameters** that you have defined:

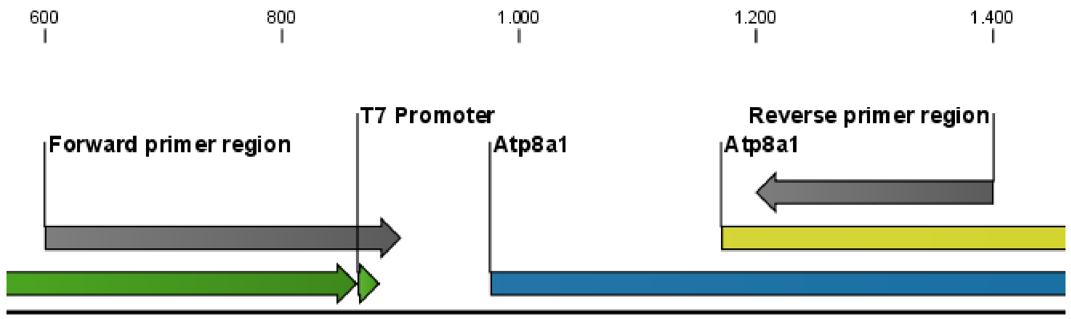
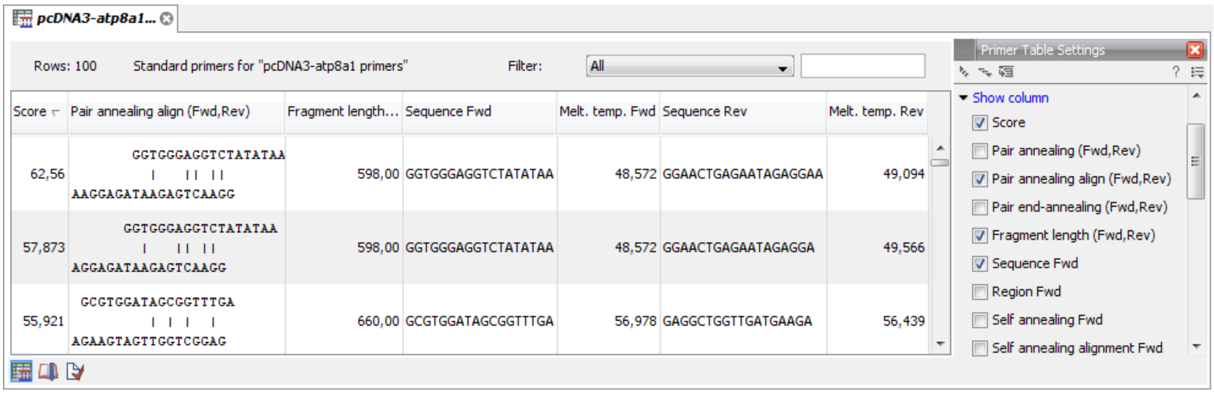


Figure 2.156: *A forward and a reverse primer region.*

Click the Calculate button | Modify parameters regarding the combination of the primers (for now, just leave them unchanged)| Calculate

This will open a table showing the possible combinations of primers. To the right, you can specify the information you want to display, e.g. showing **Fragment length** (see figure 2.157):


pcDNA3-atp8a1...

Rows: 100 Standard primers for "pcDNA3-atp8a1 primers" Filter: All

Score	Pair annealing align (Fwd,Rev)	Fragment length...	Sequence Fwd	Melt. temp. Fwd	Sequence Rev	Melt. temp. Rev
62,56	GGTGGGAGGTCTATATAA | || || AAGGAGATAAGAGTCAAGG	598,00	GGTGGGAGGTCTATATAA	48,572	GGAACTGAGAATAGAGGAA	49,094
57,873	GGTGGGAGGTCTATATAA | || || AGGAGATAAGAGTCAAGG	598,00	GGTGGGAGGTCTATATAA	48,572	GGAACTGAGAATAGAGGA	49,566
55,921	GCGTGGATAGCGGTTTGA | | | | AGAAGTAGTTGGTCGGAG	660,00	GCGTGGATAGCGGTTTGA	56,978	GAGGCTGGTTGATGAAGA	56,439

Primer Table Settings

Show column

☑ Score
□ Pair annealing (Fwd,Rev)
☑ Pair annealing align (Fwd,Rev)
□ Pair end-annealing (Fwd,Rev)
☑ Fragment length (Fwd,Rev)
☑ Sequence Fwd
□ Region Fwd
□ Self annealing Fwd
□ Self annealing alignment Fwd

Figure 2.157: *A list of primers. To the right are the Side Panel showing the available choices of information to display.*

Clicking a primer pair in the table will make a corresponding selection on the sequence in the view above. At this point, you can either settle on a specific primer pair or save the table for later. If you want to use e.g. the first primer pair for your experiment, right-click this primer pair in the table and save the primers.

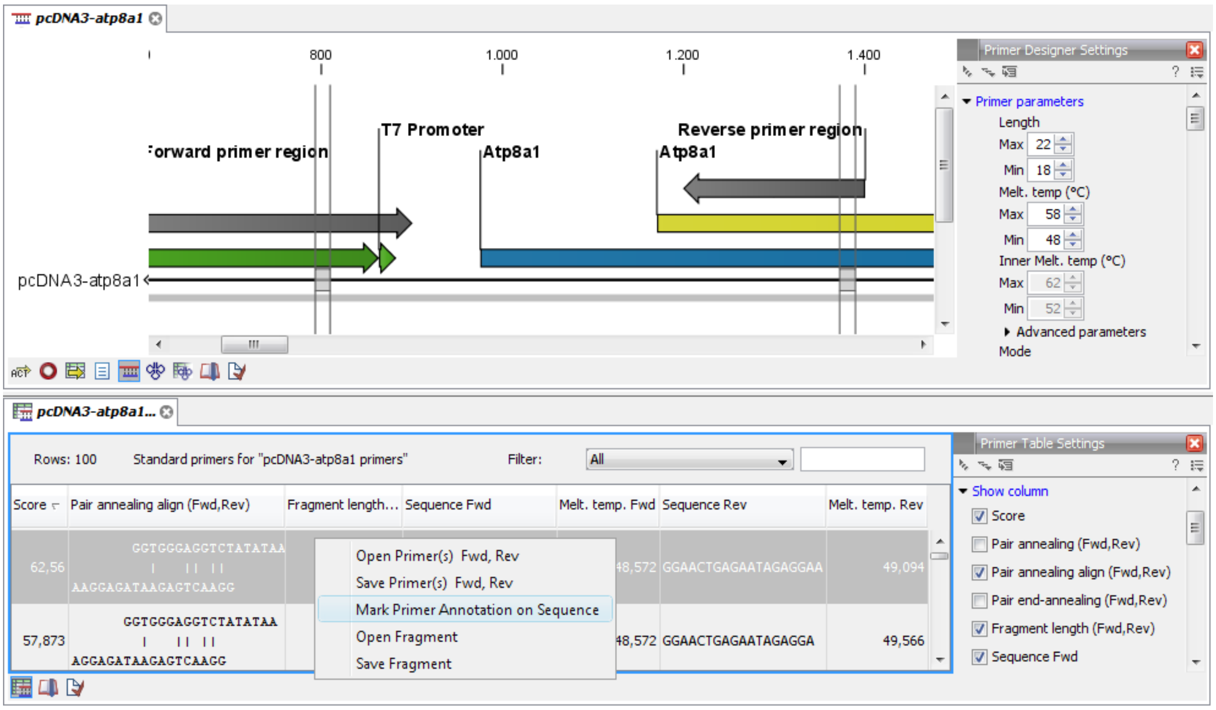
You can also mark the position of the primers on the sequence by selecting **Mark primer annotation on sequence** in the right-click menu (see figure 2.158):

You have now reached the end of this tutorial which has shown some of the many options of the primer design functionalities of *CLC Genomics Workbench*. You can read much more in the program's **Help** function (?) or in the users manual on `http://www.clcbio.com/download`.

2.22 Tutorial: BLAST search

This tutorial shows you how to perform a BLAST search using *CLC Genomics Workbench*.

Suppose you are working with the ATP8a1 protein sequence which is a phospholipid-transporting ATPase expressed in the adult house mouse, *Mus musculus*. To obtain more information about this molecule you wish to query the Swiss-Prot database to find homologous proteins in humans



Length
Max 22
Min 18
Melt. temp (°C)
Max 58
Min 48
Inner Melt. temp (°C)
Max 62
Min 52

Rows: 100 Standard primers for "pcDNA3-atp8a1 primers" Filter: All

Score	Pair annealing align (Fwd,Rev)	Fragment length...	Sequence Fwd	Melt. temp. Fwd	Sequence Rev	Melt. temp. Rev
62,56	GGTGGGAGGTCTATATAA | || || AAGGAGATAAGAGTCAAGG			48,572	GGAACTGAGAATAGAGGAA	49,094
57,873	GGTGGGAGGTCTATATAA | || || AGGAGATAAGAGTCAAGG			48,572	GGAACTGAGAATAGAGGA	49,566

☑ Score
□ Pair annealing (Fwd,Rev)
☑ Pair annealing align (Fwd,Rev)
□ Pair end-annealing (Fwd,Rev)
☑ Fragment length (Fwd,Rev)
☑ Sequence Fwd

Figure 2.158: *The options available in the right-click menu. Here, "Mark primer annotation on sequence" has been chosen, resulting in two annotations on the sequence above (labeled "Oligo").*

Homo sapiens, using the **Basic Local Alignment Search Tool** (BLAST) algorithm.

Please note that your computer must be connected to the Internet to complete this tutorial.

2.22.1 Performing the BLAST search

Start out by:

select protein ATP8a1 | **Toolbox** | **BLAST Search** ( ) | **NCBI BLAST** ( )

In **Step 1** you can choose which sequence to use as query sequence. Since you have already chosen the sequence it is displayed in the **Selected Elements** list.

Click **Next**.

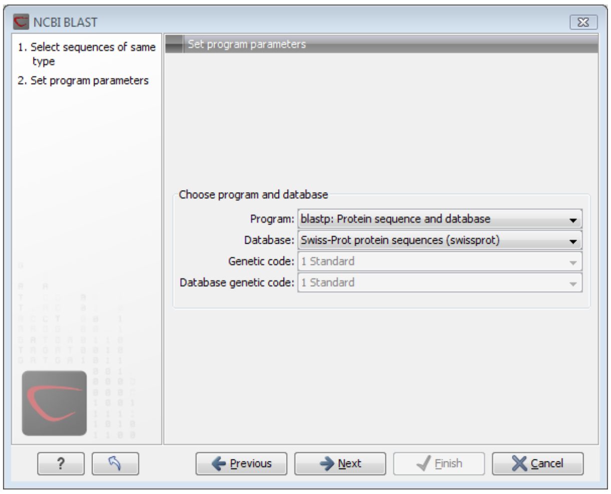
In **Step 2** (figure 2.159), choose the default BLAST program: **BLASTp: Protein sequence against Protein database** and select the **Swiss-Prot** database in the **Database** drop down menu.

Click **Next**.

In the **Limit by Entrez query** in **Step 3**, choose **Homo sapiens[ORGN]** from the drop down menu to arrive at the search configuration seen in figure 2.160. Including this term limits the query to proteins of human origin.

Click **Finish** to accept the parameter settings and begin the BLAST search.

The computer now contacts NCBI and places your query in the BLAST search queue. After a short while the result is received and opened in a new view.



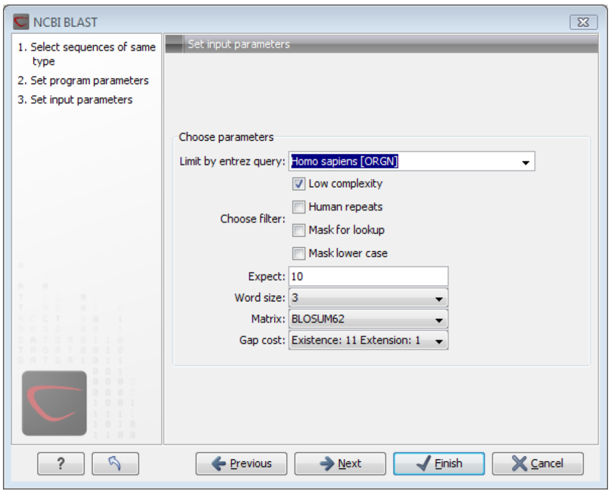
1. Select sequences of same type
2. Set program parameters

Set program parameters

Choose program and database

Program: blastp: Protein sequence and database
Database: Swiss-Prot protein sequences (swissprot)
Genetic code: 1 Standard
Database genetic code: 1 Standard

Figure 2.159: *Choosing BLAST program and database.*



1. Select sequences of same type
2. Set program parameters
3. Set input parameters

Set input parameters

Choose parameters

Limit by entrez query: Homo sapiens [ORGN]

Choose filter:
☑ Low complexity
□ Human repeats
□ Mask for lookup
□ Mask lower case

Expect: 10
Word size: 3
Matrix: BLOSUM62
Gap cost: Existence: 11 Extension: 1

Figure 2.160: *The BLAST search is limited to homo sapiens[ORGN]. The remaining parameters are left as default.*

2.22.2 Inspecting the results

The output is shown in figure 2.161 and consists of a list of potential homologs that are sorted by their BLAST match-score and shown in descending order below the query sequence.

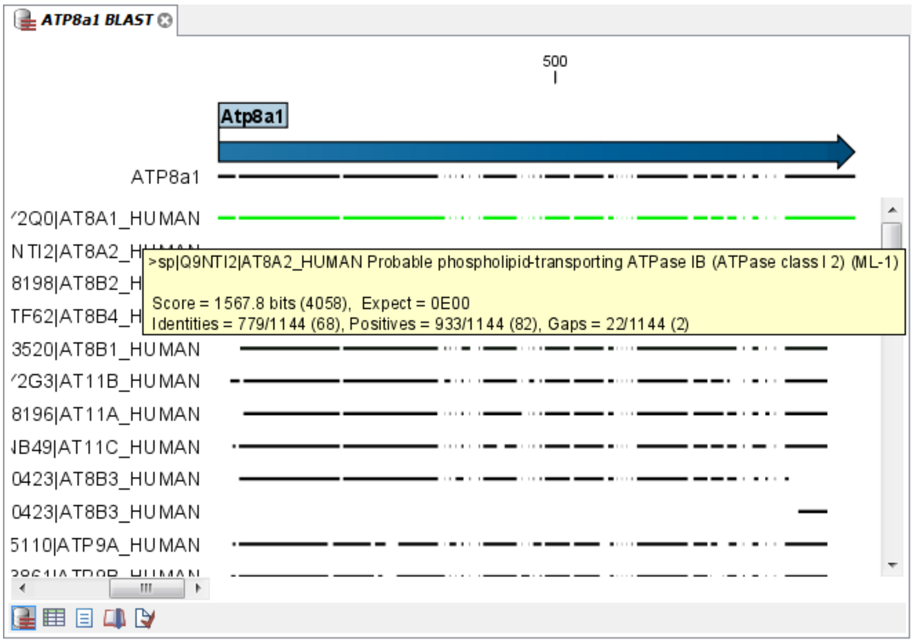
Try placing your mouse cursor over a potential homologous sequence. You will see that a context box appears containing information about the sequence and the match-scores obtained from the BLAST algorithm.

The lines in the BLAST view are the actual sequences which are downloaded. This means that you can zoom in and see the actual alignment:

> **Zoom in in the Tool Bar (🔍) | Click in the BLAST view a number of times until you see the residues**

Now we will focus our attention on sequence Q9Y2Q0 - the BLAST hit that is at the top of the list. To download the full sequence:

> **right-click the line representing sequence Q9Y2Q0 | Download Full Hit Sequence from NCBI**



>sp|Q9NTI2|AT8A2_HUMAN Probable phospholipid-transporting ATPase IB (ATPase class I 2) (ML-1)

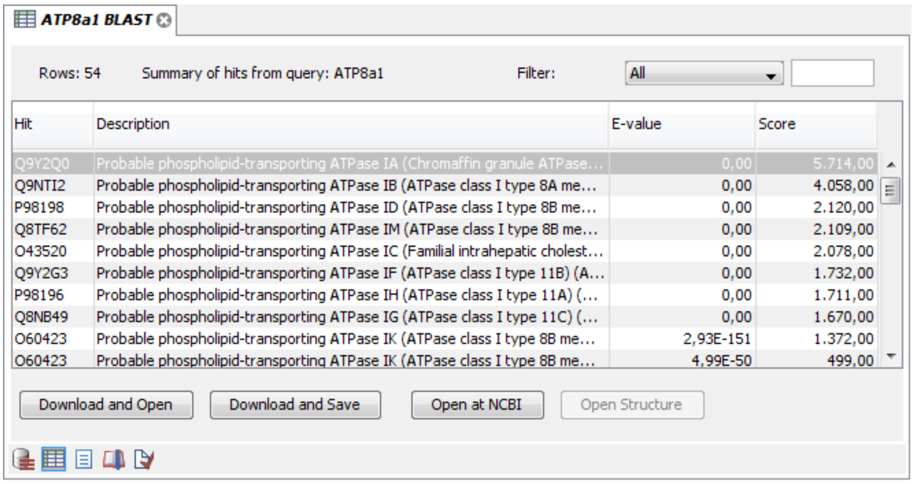
Score = 1567.8 bits (4058), Expect = 0E00
Identities = 779/1144 (68), Positives = 933/1144 (82), Gaps = 22/1144 (2)

Figure 2.161: *Output of a BLAST search. By holding the mouse pointer over the lines you can get information about the sequence.*

This opens the sequence. However, the sequence is not saved yet. Drag and drop the sequence into the **Navigation Area** to save it. This homologous sequence is now stored in the *CLC Genomics Workbench* and you can use it to gain information about the query sequence by using the various tools of the workbench, e.g. by studying its textual information, by studying its annotation or by aligning it to the query sequence.

2.22.3 Using the BLAST table view

As an alternative to the graphic BLAST view, you can click the Table View (▦) at the bottom. This will display a tabular view of the BLASt hits as shown in figure 2.162.


ATP8a1 BLAST

Rows: 54 Summary of hits from query: ATP8a1 Filter: All

Hit	Description	E-value	Score
Q9Y2Q0	Probable phospholipid-transporting ATPase IA (Chromaffin granule ATPase...	0,00	5.714,00
Q9NTI2	Probable phospholipid-transporting ATPase IB (ATPase class I type 8A me...	0,00	4.058,00
P98198	Probable phospholipid-transporting ATPase ID (ATPase class I type 8B me...	0,00	2.120,00
Q8TF62	Probable phospholipid-transporting ATPase IM (ATPase class I type 8B me...	0,00	2.109,00
O43520	Probable phospholipid-transporting ATPase IC (Familial intrahepatic cholest...	0,00	2.078,00
Q9Y2G3	Probable phospholipid-transporting ATPase IF (ATPase class I type 11B) (A...	0,00	1.732,00
P98196	Probable phospholipid-transporting ATPase IH (ATPase class I type 11A) (...	0,00	1.711,00
Q8NB49	Probable phospholipid-transporting ATPase IG (ATPase class I type 11C) (...	0,00	1.670,00
O60423	Probable phospholipid-transporting ATPase IK (ATPase class I type 8B me...	2,93E-151	1.372,00
O60423	Probable phospholipid-transporting ATPase IK (ATPase class I type 8B me...	4,99E-50	499,00

Figure 2.162: *Output of a BLAST search shown in a table.*

This view provides more statistics about the hits, and you can use the filter to search for e.g. a specific type of protein etc. If you wish to download several of the hit sequences, this is easily done in this view. Simply select the relevant sequences and drag them into a folder in the **Navigation Area**.

2.23 Tutorial: Tips for specialized BLAST searches

BLAST is a great and invaluable tool in bioinformatics. BLAST has become a very central tool when it comes to identification of homologues and similar sequences, and at the same time BLAST has evolved to become a highly complex tool which can be used for many different purposes.

In this tutorial you will learn how to:

- Use BLAST to find the gene of a protein on a genomic sequence.
- Find primer binding sites on genomic sequences
- Identify remote protein homologues.

This tutorial requires some experience using the Workbench, so if you get stuck at some point, we recommend going through the more basic tutorials first.

2.23.1 Locate a protein sequence on the chromosome

If you have a protein sequence but want to see the actual location on the chromosome this is easy to do using BLAST.

In this example we wish to map the protein sequence of the Human beta-globin protein to a chromosome. We know in advance that the beta-globin is located somewhere on chromosome 11.

Data used in this example can be downloaded from GenBank:

Search | **Search for Sequences at NCBI ()**

Human chromosome 11 (NC_000011) consists of 134452384 nucleotides and the beta-globin (AAA16334) protein has 147 amino acids.

BLAST configuration

Next, conduct a local BLAST search:

Toolbox | **BLAST Search ()** | **Local BLAST ()**

Select the protein sequence as query sequence and click **Next**. Since you wish to BLAST a protein sequence against a nucleotide sequence, use **tblastn** which will automatically translate the nucleotide sequence selected as database.

As **Target** select *NC_000011* that you downloaded. If you are used to BLAST, you will know that you usually have to create a BLAST database before BLASTing, but the Workbench does this "on the fly" when you just select one or more sequences.

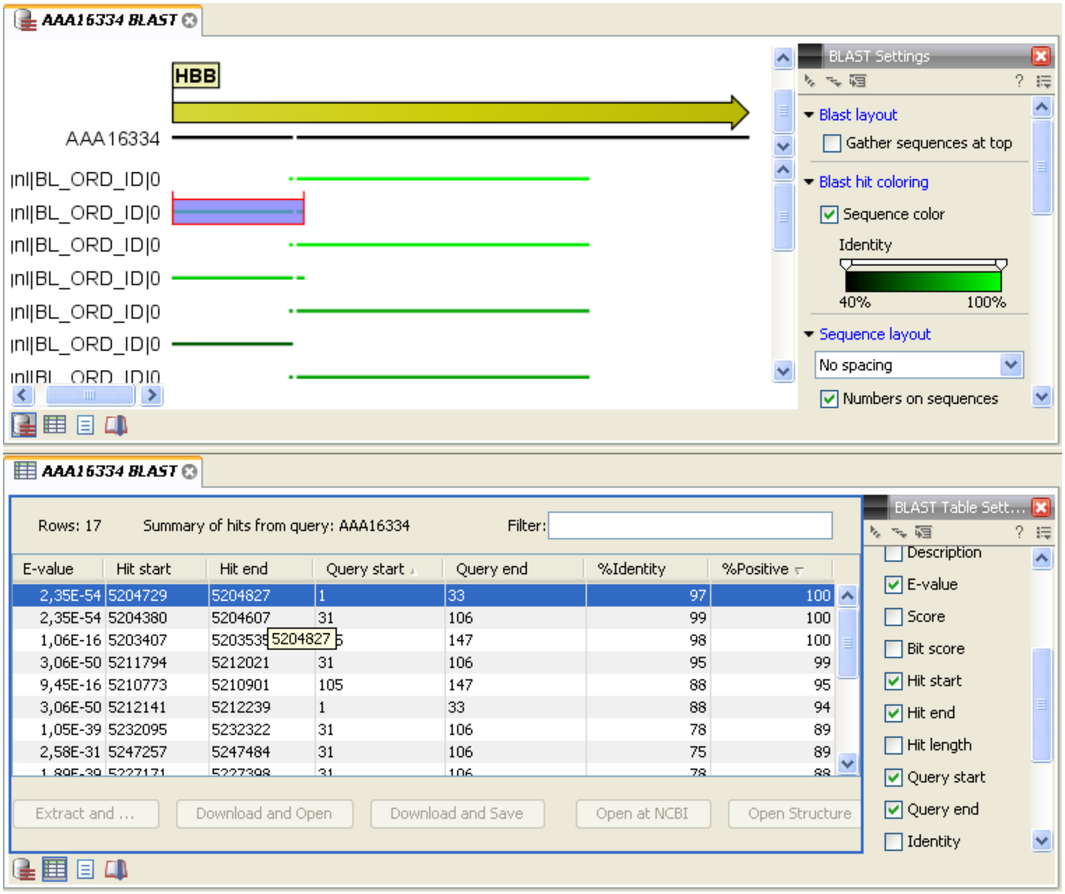
Click **Next**, leave the parameters at their default, click **Next** again, and then **Finish**.

Inspect BLAST result

When the BLAST result appears make a split view so that both the table and graphical view is visible (see figure 2.163). This is done by pressing Ctrl (⌘ on Mac) while clicking the table view (▦) at the bottom of the view.

In the table start out by showing two additional columns; "% Positive" and "Query start". These should simply be checked in the Side Panel.

Now, sort the BLAST table view by clicking the column header "% Positive". Then, press and hold the Ctrl button (⌘ on Mac) and click the header "Query start". Now you have sorted the table first on % Positive hits and then the start position of the query sequence. Now you see that you actually have three regions with a 100% positive hit but at different locations on the chromosome sequence (see figure 2.163).



▼ Blast layout
☐ Gather sequences at top
▼ Blast hit coloring
☑ Sequence color
Identity
▼ Sequence layout
No spacing
☑ Numbers on sequences

Rows: 17 Summary of hits from query: AAA16334 Filter:

E-value	Hit start	Hit end	Query start	Query end	%Identity	%Positive
2,35E-54	5204729	5204827	1	33	97	100
2,35E-54	5204380	5204607	31	106	99	100
1,06E-16	5203407	5203539 5204827	5	147	98	100
3,06E-50	5211794	5212021	31	106	95	99
9,45E-16	5210773	5210901	105	147	88	95
3,06E-50	5212141	5212239	1	33	88	94
1,05E-39	5232095	5232322	31	106	78	89
2,58E-31	5247257	5247484	31	106	75	89
1,89E-39	5227171	5227398	31	106	78	88

☐ Description
☑ E-value
☐ Score
☐ Bit score
☑ Hit start
☑ Hit end
☐ Hit length
☑ Query start
☑ Query end
☐ Identity

Figure 2.163: *Placement of translated nucleotide sequence hits on the Human beta-globin.*

Why did we find, on the protein level, three identical regions between our query protein sequence and nucleotide database?

The beta-globin gene is known to have three exons and this is exactly what we find in the BLAST search. Each translated exon will hit the corresponding sequence on the chromosome.

If you place the mouse cursor on the sequence hits in the graphical view, you can see the reading frame which is -1, -2 and -3 for the three hits, respectively.

Verify the result

Open NC_000011 in a view, and go to the Hit start position (5,204,729) and zoom to see the blue gene annotation. You can now see the exon structure of the Human beta-globin gene showing the three exons on the reverse strand (see figure 2.164).

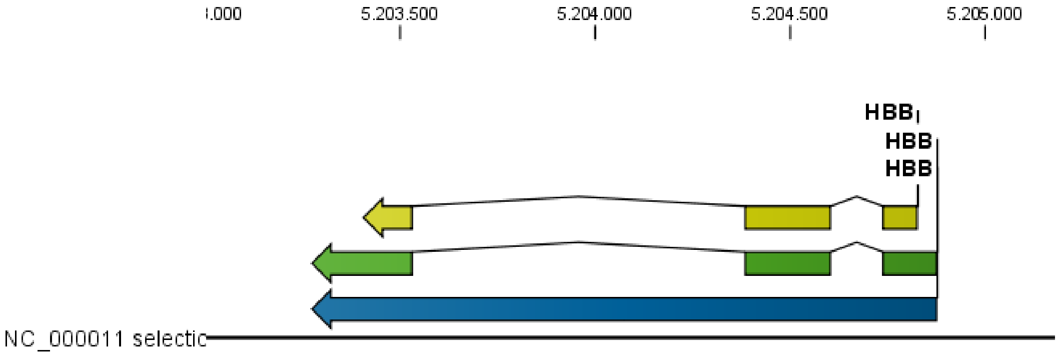


Figure 2.164: *Human beta-globin exon view.*

If you wish to verify the result, make a selection covering the gene region and open it in a new view:

right-click | Open Selection in New View ( ) | Save ( )

Save the sequence, and perform a new BLAST search:

- Use the new sequence as query.
- Use BLASTx
- Use the protein sequence, AAA16334, as database

Using the genomic sequence as query, the mapping of the protein sequence to the exons is visually very clear as shown in figure 2.165.

In theory you could use the chromosome sequence as query, but the performance would not be optimal: it would take a long time, and the computer might run out of memory.

In this example, you have used well-annotated sequences where you could have searched for the name of the gene instead of using BLAST. However, there are other situations where you either do not know the name of the gene, or the genomic sequence is poorly annotated. In these cases, the approach described in this tutorial can be very productive.

2.23.2 BLAST for primer binding sites

You can adjust the BLAST parameters so it becomes possible to match short primer sequences against a larger sequence. Then it is easy to examine whether already existing lab primers can be reused for other purposes, or if the primers you designed are specific.

Purpose	Program	Word size	Low complexity filter	Expect value
Standard BLAST	blastn	11	On	10
Primer search	blastn	7	Off	1000

These settings are shown in figure 2.166.

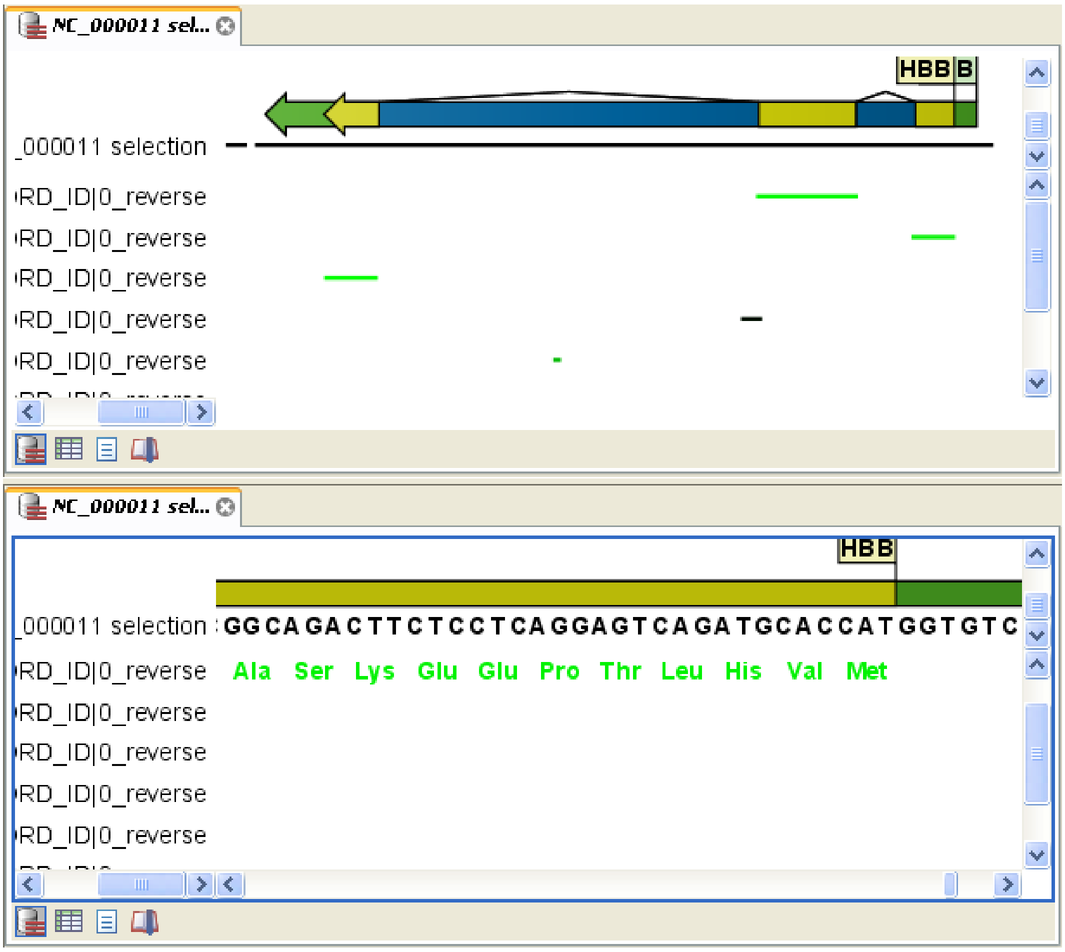


Figure 2.165: *Verification of the result: at the top a view of the whole BLAST result. At the bottom the same view is zoomed in on exon 3 to show the amino acids.*

2.23.3 Finding remote protein homologues

If you look for short identical peptide sequences in a database, the standard BLAST parameters will have to be reconfigured. Using the parameters described below, you are likely to be able to identify whether antigenic determinants will cross react to other proteins.

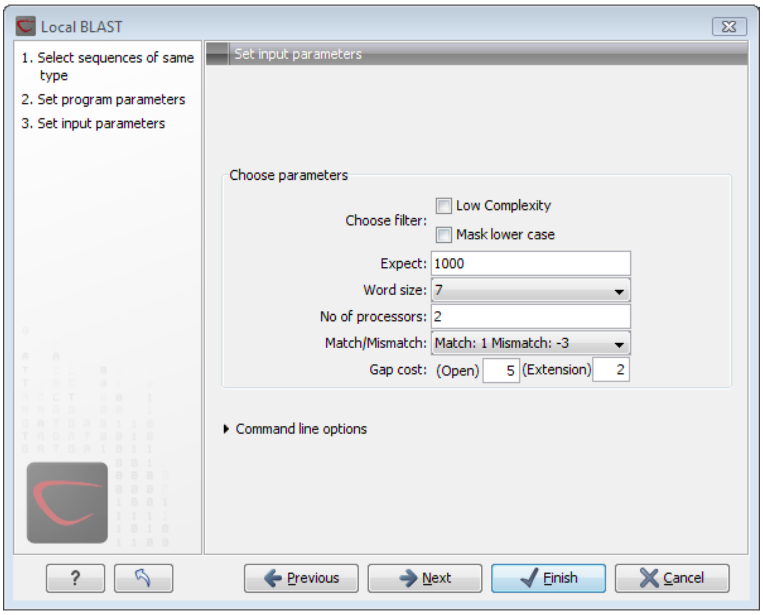
Purpose	Program	Word size	Low complexity filter	Expect value	Scoring matrix
Standard BLAST	blastp	3	On	10	BLSUM62
Remote homologues	blastp	2	Off	20000	PAM30

These settings are shown in figure 2.167.

2.23.4 Further reading

A valuable source of information about BLAST can be found at http://www.ncbi.nlm.nih.gov/blast/producttable.shtml.

Remember that BLAST is a heuristic method thus you cannot trusted BLAST to be accurate. For very accurate results you should use Smith-Waterman. You can read "Bioinformatics explained: BLAST versus Smith-Waterman" here: http://www.clcbio.com/BE.



1. Select sequences of same type
2. Set program parameters
3. Set input parameters

Set input parameters

Choose parameters

Choose filter: □ Low Complexity □ Mask lower case

Expect: 1000

Word size: 7

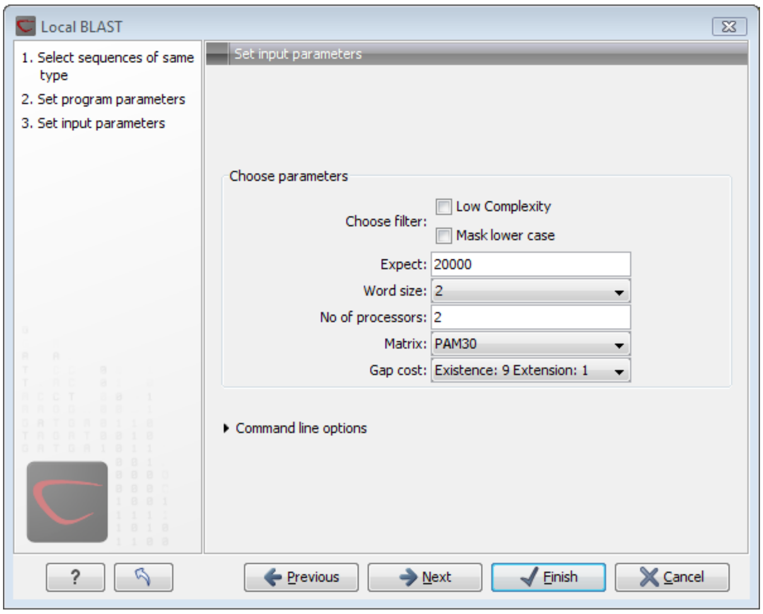
No of processors: 2

Match/Mismatch: Match: 1 Mismatch: -3

Gap cost: (Open) 5 (Extension) 2

Command line options

Figure 2.166: *Settings for searching for primer binding sites.*



1. Select sequences of same type
2. Set program parameters
3. Set input parameters

Set input parameters

Choose parameters

Choose filter: □ Low Complexity □ Mask lower case

Expect: 20000

Word size: 2

No of processors: 2

Matrix: PAM30

Gap cost: Existence: 9 Extension: 1

Command line options

Figure 2.167: *Settings for searching for remote homologues.*

2.24 Tutorial: Proteolytic cleavage detection

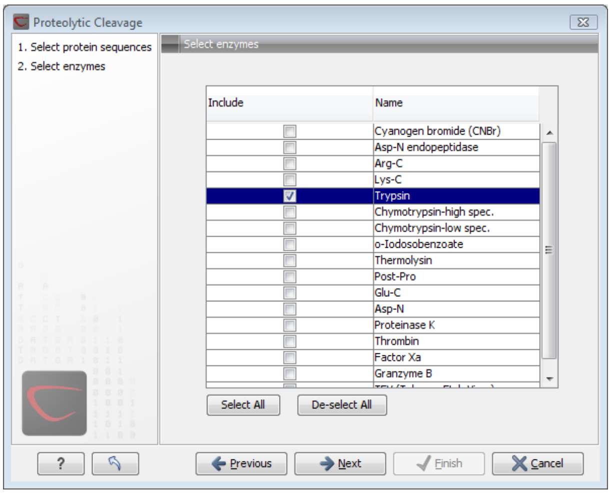
This tutorial shows you how to find cut sites and see an overview of fragments when cleaving proteins with proteolytic cleavage enzymes.

Suppose you are working with protein ATP8a1 from the example data, and you wish to see where the enzyme *trypsin* will cleave the protein. Furthermore, you want to see details for the resulting fragments which are between 10 and 15 amino acids long.

select protein ATP8a1 | **Toolbox** | **Protein Analyses ()** | **Proteolytic Cleavage**

This opens **Step 1** of the Proteolytic Cleavage dialog. In this step you can choose which sequences to include in the analysis. Since you have already chosen ATP8a1, click **Next**.

In this step you should select **Trypsin**. This is illustrated in figure 2.168.



Include	Name
□	Cyanogen bromide (CNBr)
□	Asp-N endopeptidase
□	Arg-C
□	Lys-C
☑	Trypsin
□	Chymotrypsin-high spec.
□	Chymotrypsin-low spec.
□	o-Iodosobenzoate
□	Thermolysin
□	Post-Pro
□	Glu-C
□	Asp-N
□	Proteinase K
□	Thrombin
□	Factor Xa
□	Granzyme B

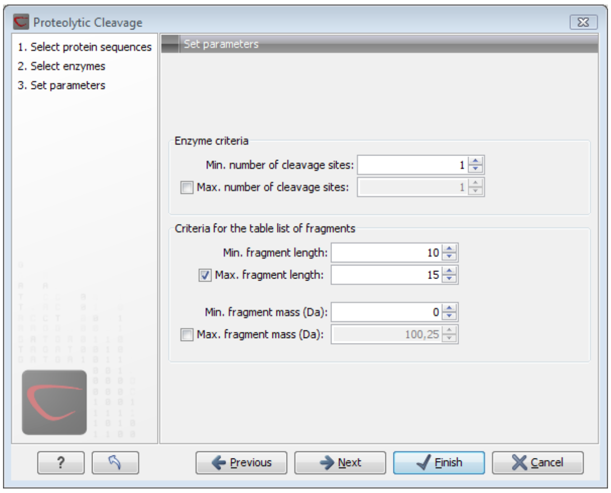
Figure 2.168: *Selecting trypsin as the cleaving enzyme.*

Click **Next** to go to **Step 3** of the dialog.

In **Step 3** you can adjust the parameters for which fragments of the cleavage you want to include in the table output of the analysis.

> **Type '10' in the Min. fragment length | Check the box: Max. fragment length | enter '15' in the corresponding text field**

These parameter adjustments are shown in figure 2.169:



Enzyme criteria

Min. number of cleavage sites: 1

□ Max. number of cleavage sites: 1

Criteria for the table list of fragments

Min. fragment length: 10

☑ Max. fragment length: 15

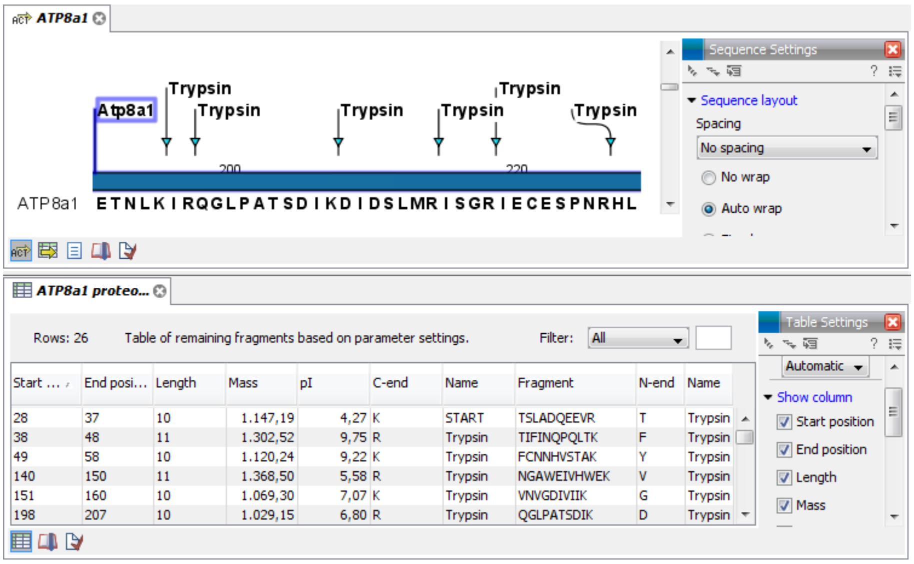
Min. fragment mass (Da): 0

□ Max. fragment mass (Da): 100,25

Figure 2.169: *Adjusting the output from the cleavage to include fragments which are between 10 and 15 amino acids long.*

Click **Finish** to make the analysis. The result of the analysis can be seen in figure 2.170

Note! The output of proteolytic cleavage is two related views. The sequence view displays annotations where the sequence is cleaved. The table view shows information about the fragments satisfying the parameters set in the dialog. Subsequently, if you have restricted the fragment parameters, you might have more annotations on the sequence than fragments in the table.



Rows: 26 Table of remaining fragments based on parameter settings. Filter: All

Start ...	End posi...	Length	Mass	pI	C-end	Name	Fragment	N-end	Name
28	37	10	1.147,19	4,27	K	START	TSLADQEEVR	T	Trypsin
38	48	11	1.302,52	9,75	R	Trypsin	TIFINQPQLTK	F	Trypsin
49	58	10	1.120,24	9,22	K	Trypsin	FCNNHVSTAK	Y	Trypsin
140	150	11	1.368,50	5,58	R	Trypsin	NGAWEIVHWEK	V	Trypsin
151	160	10	1.069,30	7,07	K	Trypsin	VNVGDIVIIK	G	Trypsin
198	207	10	1.029,15	6,80	R	Trypsin	QGLPATSDIK	D	Trypsin

☑ Start position
☑ End position
☑ Length
☑ Mass

Figure 2.170: *The output of the proteolytic cleavage shows the cleavage sites as annontations in the protein sequence. The accompanying table lists all the fragments which are between 10 and 15 amino acids long.*

If you conduct another proteolytic cleavage on the same sequence, the output consists of: (possibly) new annotations on the original sequence and an additional table view, listing all fragments.

2.25 Tutorial: Folding RNA molecules

In this tutorial, you will learn how to predict the secondary structure of an RNA molecule. You will also learn how to use the powerful ways of viewing and interacting with graphical displays of the structure.

The sequence to be folded in this tutorial is a tRNA molecule with the characteristic secondary structure as shown in figure 2.171.

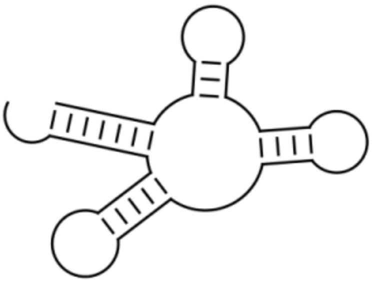


Figure 2.171: *Secondary structure of a tRNA molecule.*

The goal for this tutorial is to get a nice-looking graphic result of this structure.

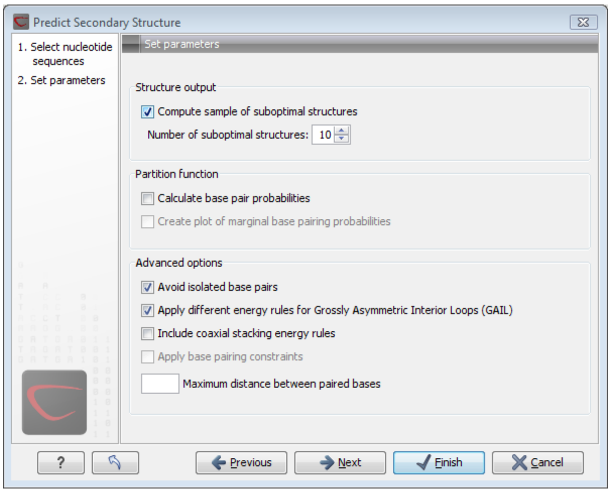
The sequence we are working with is a mitochondrial tRNA molecule from *Drosophilia melanogaster*. The name is *AB009835*, and can be found be searching GenBank:

Search | **Search for Sequences at NCBI** ( )

When you have downloaded the sequence from NCBI:

Select the sequence AB009835 | Toolbox | RNA Structure ( ) | Predict Secondary Structure ( )

Since the sequence is already selected, click **Next**. In this dialog, choose to compute a sample of sub-optimal structure and leave the rest of the settings at their default (see figure 2.172).



Predict Secondary Structure

1. Select nucleotide sequences
2. Set parameters

Set parameters

Structure output

☑ Compute sample of suboptimal structures

Number of suboptimal structures: 10

Partition function

☐ Calculate base pair probabilities

☐ Create plot of marginal base pairing probabilities

Advanced options

☑ Avoid isolated base pairs

☑ Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL)

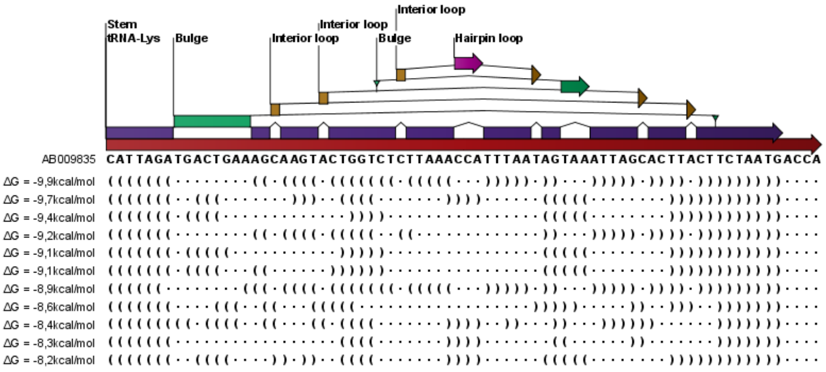
☐ Include coaxial stacking energy rules

☐ Apply base pairing constraints

Maximum distance between paired bases

Figure 2.172: *Selecting to compute 10 suboptimal structures.*

Click **Finish** and you will see a linear view of the sequence with structure information for the ten structures below the sequence, and the elements of the best structure are shown as annotations above the sequence (see figure 2.173).



```
AB009835            CATTAGATGACTGAAAGCAAGTACTGGTCTCTTAAACCATTTAATAGTAAATTAGCACTTACTTCTAATGACCA
ΔG = -9,9kcal/mol   (((((((........((.((((.(((((((.(((((...))))).))...))))).)))).)))))))))....
ΔG = -9,7kcal/mol   (((((((..(((.......)))..((((.......))))......(((((........))))))))))))....
ΔG = -9,4kcal/mol   (((((((.((((.............))))................(((((........))))))))))))....
ΔG = -9,2kcal/mol   (((((((........((.((((.(((((..((.............))...))))).)))).)))))))))....
ΔG = -9,1kcal/mol   (((((((.(((((...........)))))................(((((........))))))))))))....
ΔG = -9,1kcal/mol   (((((((.((((...((......))))))................(((((........))))))))))))....
ΔG = -8,9kcal/mol   ((((((........(((.((((.(((((((.(((((...))))).))...))))).)))).)))))))))....
ΔG = -8,6kcal/mol   (((((((....(((..((...(((((..................)))))....))..)))..)))))))....
ΔG = -8,4kcal/mol   (((((((((.((((..((...((.((((.......))))..))...))...))))))......)))))))....
ΔG = -8,3kcal/mol   (((((((.................((((.......)))).......((.......))......)))))))....
ΔG = -8,2kcal/mol   (((((((..((((....)).))..((((.......))))......(((((........))))))))))))....
```

Figure 2.173: *The inital, linear view of the secondary structure prediction.*

For now, we are not interested in the linear view. Click the **Show Secondary Structure 2D View** ( ) button at the bottom of the view to show the secondary structure. It looks as shown in figure 2.174).

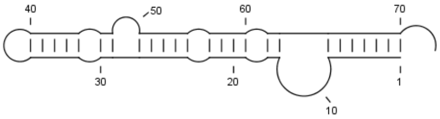
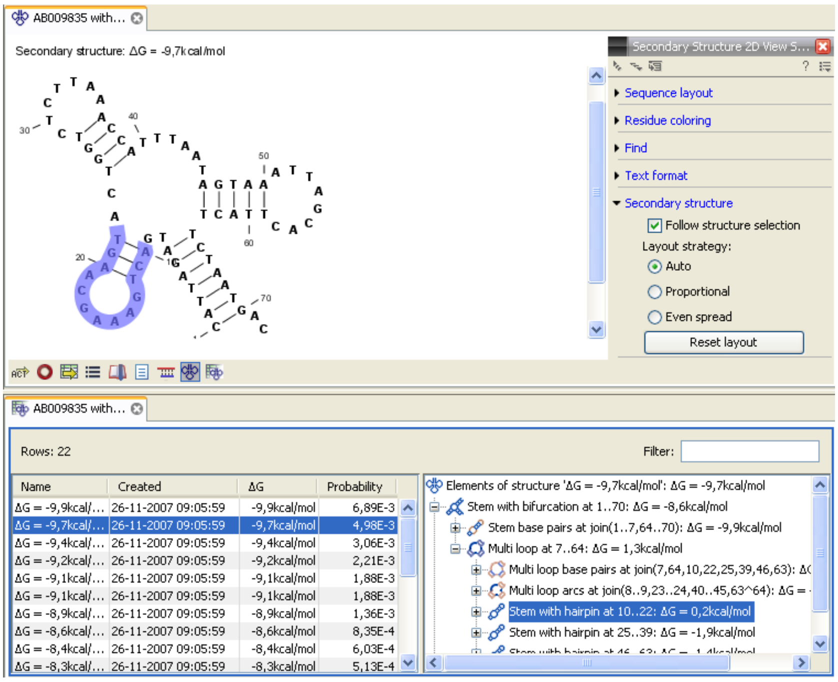


Figure 2.174: *The inital 2D view of the secondary structure.*

This structure does not look like the one we expected (shown in figure 2.171). We now take a look at some of the other structures (we chose to compute 10 different structures) to see if we can find the classic tRNA structure. First, open a split view of the **Show Secondary Structure Table** ( ):

Press and hold Ctrl (⌘ on Mac) | Show Secondary Structure Table ( )

You will now see a table displaying the ten structures. Selecting a structure in the table will display this structure in the view above. Select the second structure in the table. The views should now look like figure 2.175).



Secondary structure
☑ Follow structure selection
Layout strategy:
● Auto
○ Proportional
○ Even spread

Rows: 22

Filter:

Name	Created	ΔG	Probability
ΔG = -9,9kcal/...	26-11-2007 09:05:59	-9,9kcal/mol	6,89E-3
ΔG = -9,7kcal/...	26-11-2007 09:05:59	-9,7kcal/mol	4,98E-3
ΔG = -9,4kcal/...	26-11-2007 09:05:59	-9,4kcal/mol	3,06E-3
ΔG = -9,2kcal/...	26-11-2007 09:05:59	-9,2kcal/mol	2,21E-3
ΔG = -9,1kcal/...	26-11-2007 09:05:59	-9,1kcal/mol	1,88E-3
ΔG = -9,1kcal/...	26-11-2007 09:05:59	-9,1kcal/mol	1,88E-3
ΔG = -8,9kcal/...	26-11-2007 09:05:59	-8,9kcal/mol	1,36E-3
ΔG = -8,6kcal/...	26-11-2007 09:05:59	-8,6kcal/mol	8,35E-4
ΔG = -8,4kcal/...	26-11-2007 09:05:59	-8,4kcal/mol	6,03E-4
ΔG = -8,3kcal/...	26-11-2007 09:05:59	-8,3kcal/mol	5,13E-4

Elements of structure 'ΔG = -9,7kcal/mol': ΔG = -9,7kcal/mol
- Stem with bifurcation at 1..70: ΔG = -8,6kcal/mol
  - Stem base pairs at join(1..7,64..70): ΔG = -9,9kcal/mol
  - Multi loop at 7..64: ΔG = 1,3kcal/mol
    - Multi loop base pairs at join(7,64,10,22,25,39,46,63): ΔG
    - Multi loop arcs at join(8..9,23..24,40..45,63^64): ΔG =
    - Stem with hairpin at 10..22: ΔG = 0,2kcal/mol
    - Stem with hairpin at 25..39: ΔG = -1,9kcal/mol

Figure 2.175: *A split view showing the scondary structure table at the bottom and the Secondary structure 2D view at the top. (You might need to Zoom out to see the structure).*

The secondary structure now looks very similar to figure 2.171. By adjusting the layout, we can make it look exactly the same: in the Side Panel of the 2D view, under **Secondary Structure**, choose the **Proportional** layout strategy. You will now see that the appearance of structure changes.

Next, zoom in on the structure to see the residues. This is easiest if you first close (☒) the table view at the bottom.

Zoom in ( ) | Click the structure until you see the residues

If you wish to make some manual corrections of the layout of the structure, first select the **Pan** ( ) mode in the Tool bar. Now place the mouse cursor on the opening of a stem, and a visual indication of the anchor point for turning the substructure will be shown (see figure 24.14).

Click and drag to rotate the part of the structure represented by the line going from the anchor point. In order to keep the bases in a relatively sequential arrangement, there is a restriction on how much the substructure can be rotated. The highlighted part of the circle represents the angle where rotating is allowed.

In figure 24.15, the structure shown in figure 24.14 has been modified by dragging with the mouse.

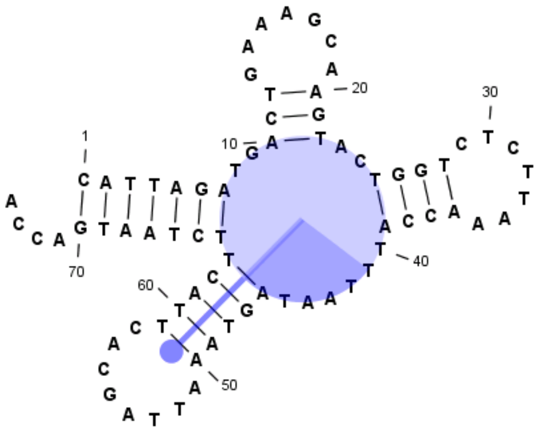


Figure 2.176: *The blue circle represents the anchor point for rotating the substructure.*

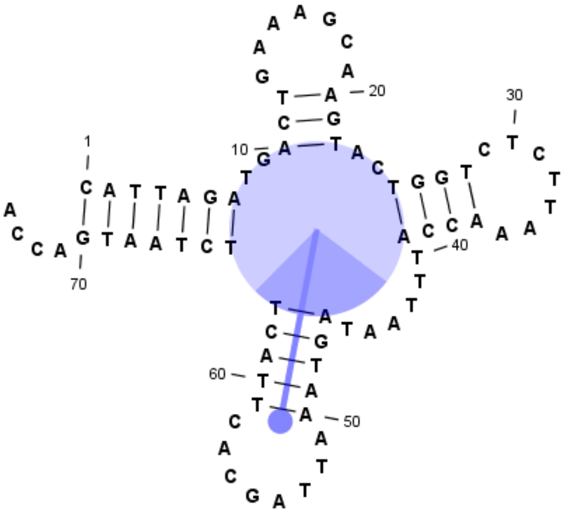


Figure 2.177: *The structure has now been rotated.*

The view can of course be printed (🖨) or exported as graphics (🖼).

2.26 Tutorial: Align protein sequences

It is possible to create multiple alignments of nucleotide and protein sequences. *CLC Genomics Workbench* offers several opportunities to view alignments. The alignments can be used for building phylogenetic trees.

The sequences must be saved in the **Navigation Area** in order to be included in an alignment. To save a sequence which is displayed in the **View Area**, click the tab of the sequence and press Ctrl + S (or ⌘ + S on Mac). In this tutorial six protein sequences from the Example data will be aligned. (See figure 2.178).

To align the sequences:

> **select the sequences from the 'Protein' folder under 'Sequences' | Toolbox | Alignments and Trees ( ) | Create Alignment ( )**

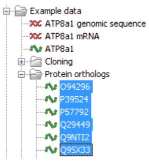


Figure 2.178: *Six protein sequences in 'Sequences' from the 'Protein orthologs' folder of the Example data.*

2.26.1 The alignment dialog

This opens the dialog shown in figure 2.179.

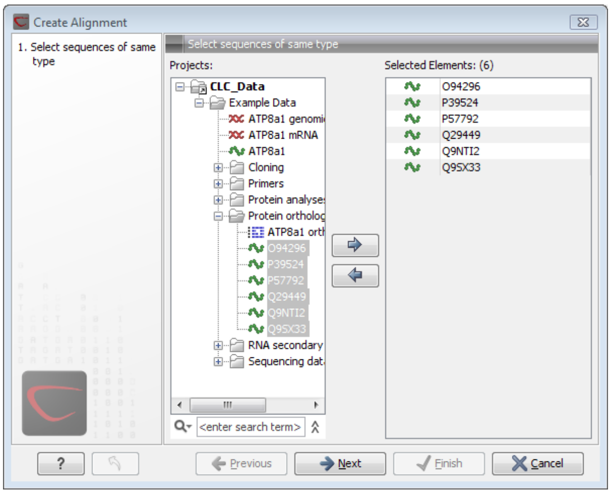
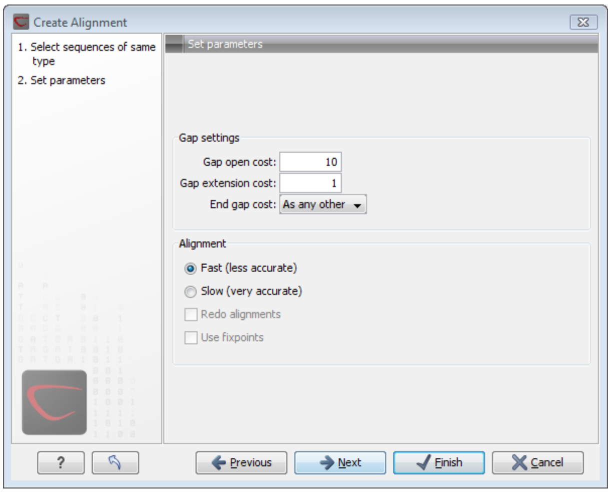


Figure 2.179: *The alignment dialog displaying the six protein sequences.*

It is possible to add and remove sequences from **Selected Elements** list. Since we had already selected the eight proteins, just click **Next** to adjust parameters for the alignment.

Clicking **Next** opens the dialog shown in figure 2.180.



Gap settings

Gap open cost: 10

Gap extension cost: 1

End gap cost: As any other

Alignment

● Fast (less accurate)

○ Slow (very accurate)

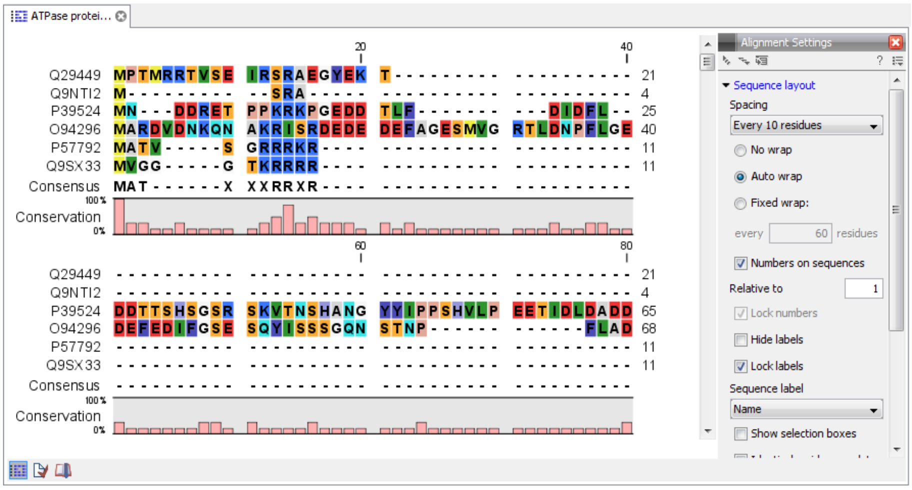
□ Redo alignments

□ Use fixpoints

Figure 2.180: *The alignment dialog displaying the available parameters which can be adjusted.*

Leave the parameters at their default settings. An explanation of the parameters can be found by clicking the help button (?). Alternatively, a tooltip is displayed by holding the mouse cursor on the parameters.

Click **Finish** to start the alignment process which is shown in the **Toolbox** under the **Processes** tab. When the program is finished calculating it displays the alignment (see fig. 2.181):



```
                                    20                  40
Q29449     MPTMRRTVSE IRSRAEGYEK T--------- ---------- 21
Q9NTI2     M--------- --SRA----- ---------- ---------- 4
P39524     MN---DDRET PPKRKPGEDD TLF------- ---DIDFL-- 25
O94296     MARDVDNKQN AKRISRDEDE DEFAGESMVG RTLDNPFLGE 40
P57792     MATV-----S GRRRKR---- ---------- ---------- 11
Q9SX33     MVGG-----G TKRRRR---- ---------- ---------- 11
Consensus  MAT------X XXRRXR---- ---------- ----------

                                    60                  80
Q29449     ---------- ---------- ---------- ---------- 21
Q9NTI2     ---------- ---------- ---------- ---------- 4
P39524     DDTTSHSGSR SKVTNSHANG YYIPPSHVLP EETIDLDADD 65
O94296     DEFEDIFGSE SQYISSSGQN STNP------ ------FLAD 68
P57792     ---------- ---------- ---------- ---------- 11
Q9SX33     ---------- ---------- ---------- ---------- 11
Consensus  ---------- ---------- ---------- ----------
```

Sequence layout

Spacing: Every 10 residues

○ No wrap
● Auto wrap
○ Fixed wrap:

every 60 residues

☑ Numbers on sequences

Relative to 1

☑ Lock numbers

☐ Hide labels

☑ Lock labels

Sequence label: Name

☐ Show selection boxes

Figure 2.181: *The resulting alignment.*

Note! The new alignment is not saved automatically.

To save the alignment, drag the tab of the alignment view into the **Navigation Area**.

If you wish to use other alignment algorithms (like e.g. ClustalW), please download the Additional Alignments Module from `http://www.clcbio.com/plugins`.

2.27 Tutorial: Create and modify a phylogenetic tree

You can make a phylogenetic tree from an existing alignment. (See how to create an alignment in the tutorial: "Align protein sequences").

We use the 'ATPase protein alignment' located in 'Protein orthologs' in the Example data. To create a phylogenetic tree:

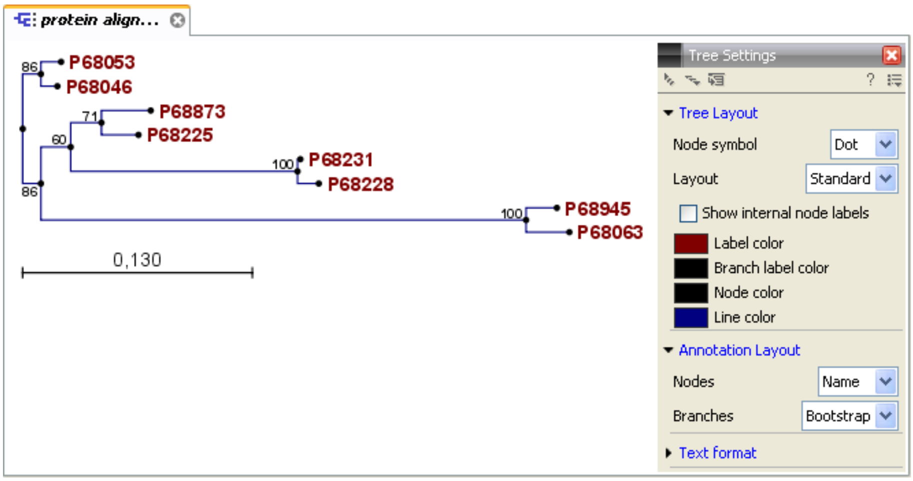
> **click the 'ATPase protein alignment' in the Navigation Area | Toolbox | Alignments and Trees ( ) | Create Tree ( )**

A dialog opens where you can confirm your selection of the alignment. Click **Next** to move to the next step in the dialog where you can choose between the neighbor joining and the UPGMA algorithms for making trees. You also have the option of including a bootstrap analysis of the result. Leave the parameters at their default, and click **Finish** to start the calculation, which can be seen in the **Toolbox** under the **Processes** tab. After a short while a tree appears in the **View Area** (figure 2.182).

2.27.1 Tree layout

Using the **Side Panel** (in the right side of the view), you can change the way the tree is displayed.

Click **Tree Layout** and open the **Layout** drop down menu. Here you can choose between standard and topology layout. The topology layout can help to give an overview of the tree if some of the branches are very short.



Tree Layout

Node symbol: Dot

Layout: Standard

□ Show internal node labels

Label color

Branch label color

Node color

Line color

Annotation Layout

Nodes: Name

Branches: Bootstrap

Text format

Figure 2.182: *After choosing which algorithm should be used, the tree appears in the View Area. The Side panel in the right side of the view allows you to adjust the way the tree is displayed.*

When the sequences include the appropriate annotation, it is possible to choose between the accession number and the species names at the leaves of the tree. Sequences downloaded from GenBank, for example, have this information. The **Labels** preferences allows these different node annotations as well as different annotation on the branches.

The branch annotation includes the bootstrap value, if this was selected when the tree was calculated. It is also possible to annotate the branches with their lengths.

2.28 Tutorial: Find restriction sites

This tutorial will show you how to find restriction sites and annotate them on a sequence.

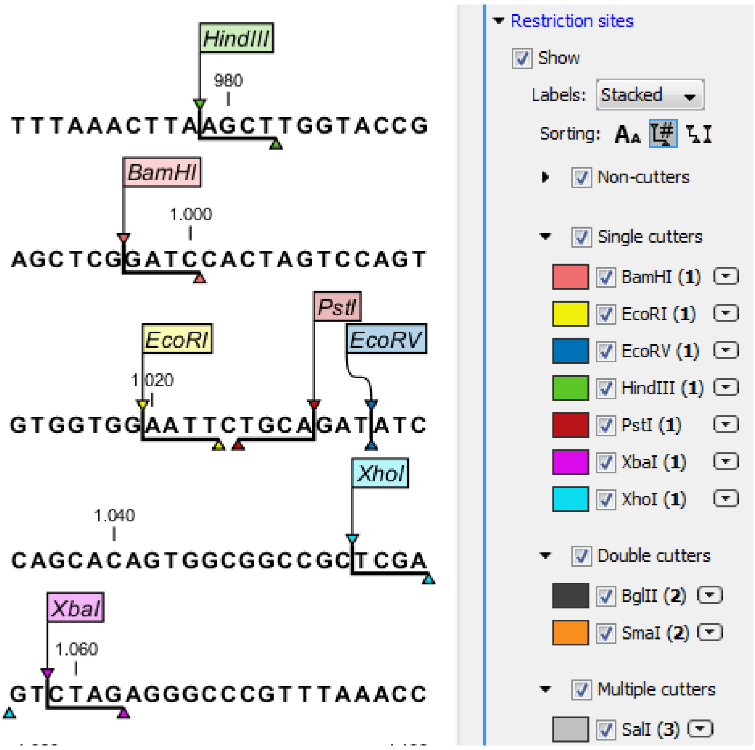
There are two ways of finding and showing restriction sites. In many cases, the dynamic restriction sites found in the **Side Panel** of sequence views will be useful, since it is a quick and easy way of showing restriction sites. In the **Toolbox** you will find the other way of doing restriction site analyses. This way provides more control of the analysis and gives you more output options, e.g. a table of restriction sites and a list of restriction enzymes that can be saved for later use. In this tutorial, the first section describes how to use the Side Panel to show restriction sites, whereas the second section describes the restriction map analysis performed from the **Toolbox**.

2.28.1 The Side Panel way of finding restriction sites

When you open a sequence, there is a **Restriction sites** setting in the **Side Panel**. By default, 10 of the most popular restriction enzymes are shown (see figure 2.183).

The restriction sites are shown on the sequence with an indication of cut site and recognition sequence. In the list of enzymes in the **Side Panel**, the number of cut sites is shown in parentheses for each enzyme (e.g. *SalI* cuts three times). If you wish to see the recognition sequence of the enzyme, place your mouse cursor on the enzyme in the list for a short moment, and a tool tip will appear.

You can add or remove enzymes from the list by clicking the **Manage enzymes** button.



▾ Restriction sites

☑ Show

Labels: Stacked

Sorting:

▸ ☑ Non-cutters

▾ ☑ Single cutters

- ☑ BamHI (1)
- ☑ EcoRI (1)
- ☑ EcoRV (1)
- ☑ HindIII (1)
- ☑ PstI (1)
- ☑ XbaI (1)
- ☑ XhoI (1)

▾ ☑ Double cutters

- ☑ BglII (2)
- ☑ SmaI (2)

▾ ☑ Multiple cutters

- ☑ SalI (3)

Figure 2.183: *Showing restriction sites of ten restriction enzymes.*

2.28.2 The Toolbox way of finding restriction sites

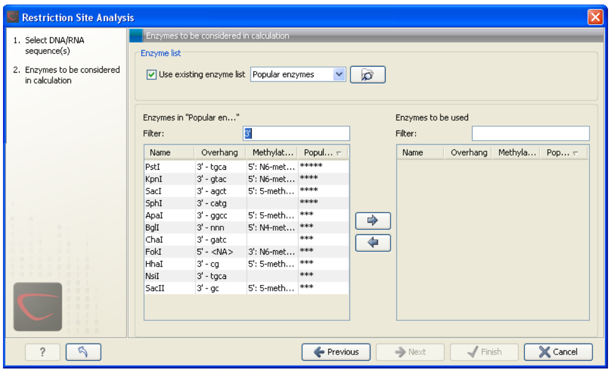
Suppose you are working with sequence 'ATP8a1 mRNA' from the example data, and you wish to know which restriction enzymes will cut this sequence exactly once and create a 3' overhang. Do the following:

> **select the ATP8a1 mRNA sequence** | **Toolbox in the Menu Bar** | **Cloning and Restriction Sites** ( ) | **Restriction Site Analysis** ( )

Click **Next** to set parameters for the restriction map analysis.

In this step first select **Use existing enzyme list** and click the **Browse for enzyme list** button ( ). Select the 'Popular enzymes' in the Cloning folder under Enzyme lists.

Then write 3' into the filter below to the left. Select all the enzymes and click the **Add** button ( ). The result should be like in figure 2.184.



1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation

Enzymes to be considered in calculation

Enzyme list

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter: 3'

Name	Overhang	Methylat...	Popul...
PstI	3' - tgca	5': N6-met...	*****
KpnI	3' - gtac	5': N6-met...	****
SacI	3' - agct	5': 5-meth...	****
SphI	3' - catg		****
ApaI	3' - ggcc	5': 5-meth...	***
BglI	3' - nnn	5': N4-met...	***
ChaI	3' - gatc		***
FokI	5' - <NA>	3': N6-met...	***
HhaI	3' - cg	5': 5-meth...	***
NsiI	3' - tgca		***
SacII	3' - gc	5': 5-meth...	***

Enzymes to be used

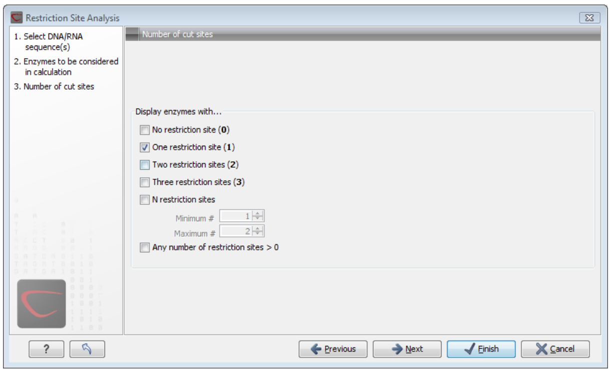
Filter:

Name	Overhang	Methyla...	Pop...

Figure 2.184: *Selecting enzymes.*

Click **Next**. In this step you specify that you want to show enzymes that cut the sequence only once. This means that you should de-select the **Two restriction sites** checkbox.

Click **Next** and select that you want to **Add restriction sites as annotations on sequence** and **Create restriction map**. (See figure 2.185).



Restriction Site Analysis

1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation
3. Number of cut sites

Number of cut sites

Display enzymes with...

- ☐ No restriction site (**0**)
- ☑ One restriction site (**1**)
- ☐ Two restriction sites (**2**)
- ☐ Three restriction sites (**3**)
- ☐ N restriction sites
  - Minimum # 1
  - Maximum # 2
- ☐ Any number of restriction sites > 0

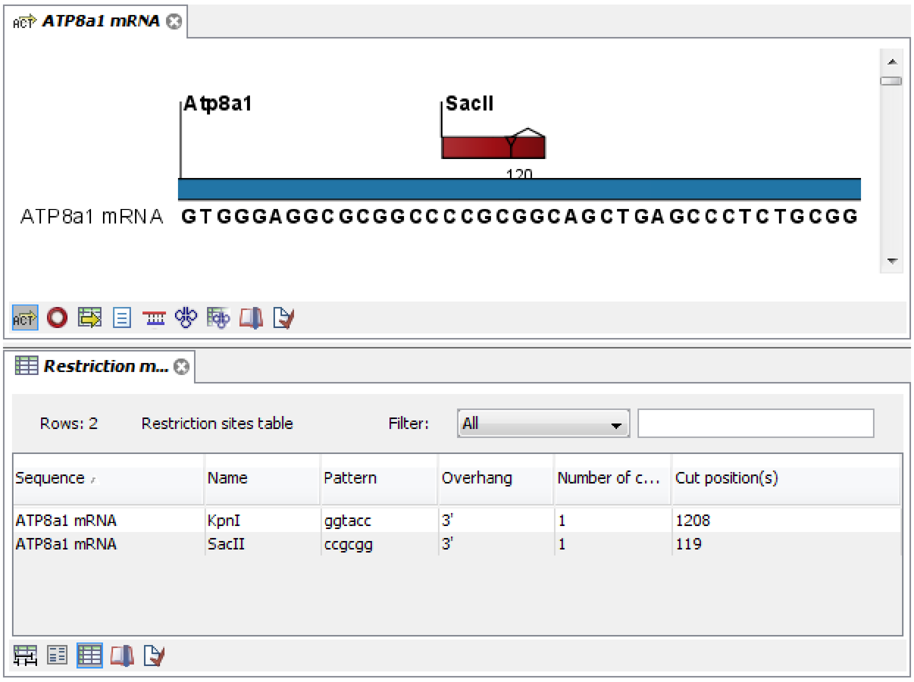
Figure 2.185: *Selecting output for restriction map analysis.*

Click **Finish** to start the restriction map analysis.

View restriction site

The restriction sites are shown in two views: one view is in a tabular format and the other view displays the sites as annotations on the sequence.

The result is shown in figure 2.186. The restriction map at the bottom can also be shown as a table of fragments produced by cutting the sequence with the enzymes:



Rows: 2 Restriction sites table Filter: All

Sequence	Name	Pattern	Overhang	Number of c...	Cut position(s)
ATP8a1 mRNA	KpnI	ggtacc	3'	1	1208
ATP8a1 mRNA	SacII	ccgcgg	3'	1	119

Figure 2.186: *The result of the restriction map analysis is displayed in a table at the bottom and as annotations on the sequence in the view at the top.*

Click the Fragments button (▤) at the bottom of the view

In a similar way the fragments can be shown on a virtual gel:

Click the Gel button (☷) at the bottom of the view

Part II

Core Functionalities

Chapter 3

User interface

Contents

This chapter provides an overview of the different areas in the user interface of *CLC Genomics Workbench*. As can be seen from figure 3.1 this includes a **Navigation Area**, **View Area**, **Menu Bar**, **Toolbar**, **Status Bar** and **Toolbox**.

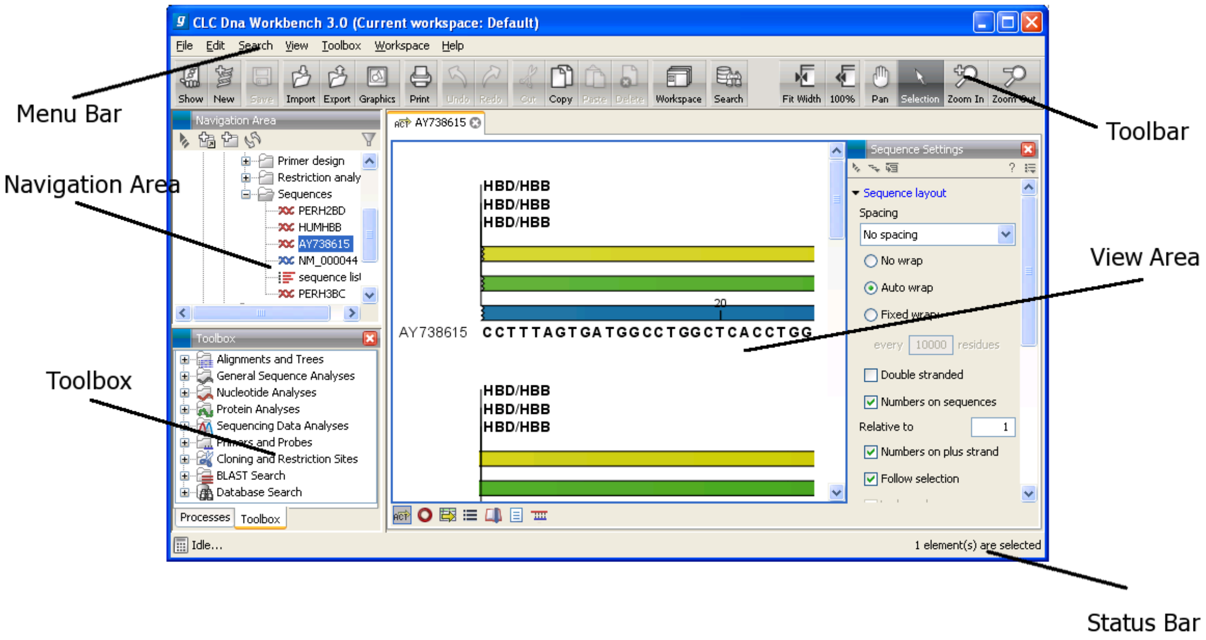


Figure 3.1: *The user interface consists of the Menu Bar, Toolbar, Status Bar, Navigation Area, Toolbox, and View Area.*

3.1 Navigation Area

The **Navigation Area** is located in the left side of the screen, under the **Toolbar** (see figure 3.2). It is used for organizing and navigating data. Its behavior is similar to the way files and folders are usually displayed on your computer.

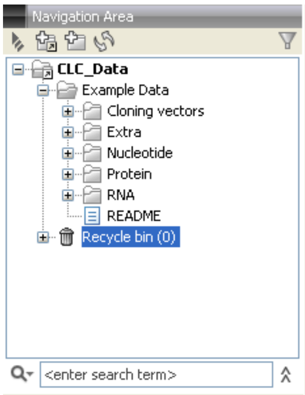


Figure 3.2: *The Navigation Area.*

3.1.1 Data structure

The data in the **Navigation Area** is organized into a number of **Locations**. When the *CLC Genomics Workbench* is started for the first time, there is one location called *CLC_Data* (unless your computer administrator has configured the installation otherwise).

A location represents a folder on the computer: The data shown under a location in the **Navigation Area** is stored on the computer in the folder which the location points to.

This is explained visually in figure 3.3.

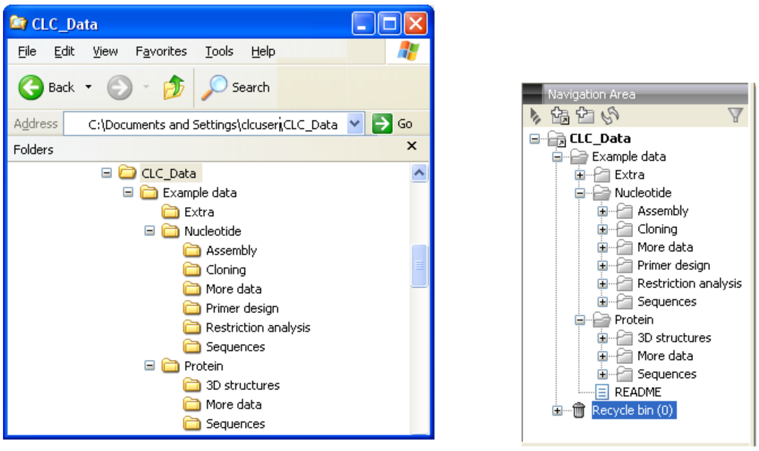


Figure 3.3: *In this example the location called 'CLC_Data' points to the folder at C:\Documents and settings\clcuser\CLC_Data.*

Adding locations

Per default, there is one location in the **Navigation Area** called CLC_Data. It points to the following folder:

- On Windows: C:\Documents and settings\<username>\CLC_Data
- On Mac: ~/CLC_Data
- On Linux: /homefolder/CLC_Data

You can easily add more locations to the **Navigation Area**:

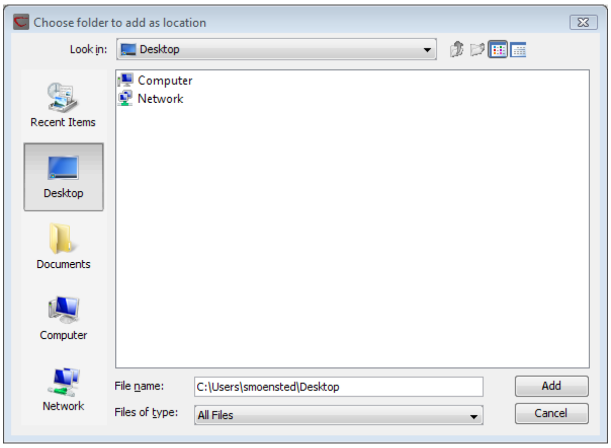
File | New | Location ()

This will bring up a dialog where you can navigate to the folder you wish to use as your new location (see figure 3.4).

When you click **Open**, the new location is added to the **Navigation Area** as shown in figure 3.5.

The name of the new location will be the name of the folder selected for the location. To see where the folder is located on your computer, place your mouse cursor on the location icon () for second. This will show the path to the location.

Sharing data is possible of you add a location on a network drive. The procedure is similar to the one described above. When you add a location on a network drive or a removable drive, the



Look in: Desktop

File name: C:\Users\smoensted\Desktop

Files of type: All Files

Figure 3.4: *Navigating to a folder to use as a new location.*

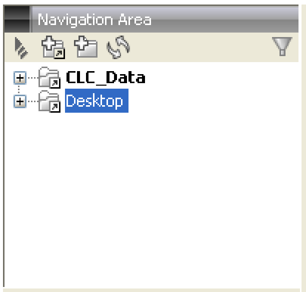


Figure 3.5: *The new location has been added.*

location will appear *inactive* when you are not connected. Once you connect to the drive again, click **Update All** () and it will become active (note that there will be a few seconds' delay from you connect).

Opening data

The elements in the **Navigation Area** are opened by :

Double-click the element

or **Click the element | Show () in the Toolbar | Select the desired way to view the element**

This will open a view in the **View Area**, which is described in section 3.2.

Adding data

Data can be added to the **Navigation Area** in a number of ways. Files can be imported from the file system (see chapter 7). Furthermore, an element can be added by dragging it into the **Navigation Area**. This could be views that are open, elements on lists, e.g. search hits or sequence lists, and files located on your computer. Finally, you can add data by adding a new location (see section 3.1.1).

If a file or another element is dropped on a folder, it is placed at the bottom of the folder. If it is dropped on another element, it will be placed just below that element.

If the element already exists in the **Navigation Area**, you will be asked whether you wish to create a copy.

3.1.2 Create new folders

In order to organize your files, they can be placed in folders. Creating a new folder can be done in two ways:

right-click an element in the Navigation Area | **New** | **Folder (📁)**

or **File** | **New** | **Folder (📁)**

If a folder is selected in the **Navigation Area** when adding a new folder, the new folder is added at the bottom of this folder. If an element is selected, the new folder is added right above that element.

You can move the folder manually by selecting it and dragging it to the desired destination.

3.1.3 Sorting folders

You can sort the elements in a folder alphabetically:

right-click the folder | **Sort Folder**

On Windows, subfolders will be placed at the top of the folder, and the rest of the elements will be listed below in alphabetical order. On Mac, both subfolders and other elements are listed together in alphabetical order.

3.1.4 Multiselecting elements

Multiselecting elements means that you select more than one element at the same time. This can be done in the following ways:

- Holding down the <Ctrl> key (⌘ on Mac) while clicking on multiple elements selects the elements that have been clicked.
- Selecting one element, and selecting another element while holding down the <Shift> key selects all the elements listed between the two locations (the two end locations included).
- Selecting one element, and moving the curser with the arrow-keys while holding down the <Shift> key, enables you to increase the number of elements selected.

3.1.5 Moving and copying elements

Elements can be moved and copied in several ways:

- Using **Copy** (⎘), **Cut** (✂) and **Paste** (📋) from the **Edit** menu.
- Using Ctrl + C (⌘ + C on Mac), Ctrl + X (⌘ + X on Mac) and Ctrl + V (⌘ + V on Mac).
- Using **Copy** (⎘), **Cut** (✂) and **Paste** (📋) in the **Toolbar**.
- Using drag and drop to move elements.

- Using drag and drop while pressing Ctrl / Command to copy elements.

In the following, all of these possibilities for moving and copying elements are described in further detail.

Copy, cut and paste functions

Copies of elements and folders can be made with the copy/paste function which can be applied in a number of ways:

select the files to copy | right-click one of the selected files | Copy (🗐) | right-click the location to insert files into | Paste (📋)

or **select the files to copy | Ctrl + C (⌘ + C on Mac) | select where to insert files | Ctrl + P (⌘ + P on Mac)**

or **select the files to copy | Edit in the Menu Bar | Copy (🗐) | select where to insert files | Edit in the Menu Bar | Paste (📋)**

If there is already an element of that name, the pasted element will be renamed by appending a number at the end of the name.

Elements can also be moved instead of copied. This is done with the cut/paste function:

select the files to cut | right-click one of the selected files | Cut (✂) | right-click the location to insert files into | Paste (📋)

or **select the files to cut | Ctrl + X (⌘ + X on Mac) | select where to insert files | Ctrl + V (⌘ + V on Mac)**

When you have cut the element, it is "greyed out" until you activate the paste function. If you change your mind, you can revert the cut command by copying another element.

Note that if you move data between locations, the original data is kept. This means that you are essentially doing a copy instead of a move operation.

Move using drag and drop

Using drag and drop in the **Navigation Area**, as well as in general, is a four-step process:

click the element | click on the element again, and hold left mouse button | drag the element to the desired location | let go of mouse button

This allows you to:

- Move elements between different folders in the **Navigation Area**
- Drag from the **Navigation Area** to the **View Area**: A new view is opened in an existing **View Area** if the element is dragged from the **Navigation Area** and dropped next to the tab(s) in that **View Area**.
- Drag from the **View Area** to the **Navigation Area**: The element, e.g. a sequence, alignment, search report etc. is saved where it is dropped. If the element already exists, you are asked whether you want to save a copy. You drag from the **View Area** by dragging the tab of the desired element.

Use of drag and drop is supported throughout the program, also to open and re-arrange views (see section 3.2.6).

Note that if you move data between locations, the original data is kept. This means that you are essentially doing a copy instead of a move operation.

Copy using drag and drop

To copy instead of move using drag and drop, hold the Ctrl (⌘ on Mac) key while dragging:

> **click the element | click on the element again, and hold left mouse button | drag the element to the desired location | press Ctrl (⌘ on Mac) while you let go of mouse button release the Ctrl/⌘ button**

3.1.6 Change element names

This section describes two ways of changing the names of sequences in the **Navigation Area**. In the first part, the sequences themselves are not changed - it's their representation that changes. The second part describes how to change the name of the element.

Change how sequences are displayed

Sequence elements can be displayed in the **Navigation Area** with different types of information:

- Name (this is the default information to be shown).
- Accession (sequences downloaded from databases like GenBank have an accession number).
- Latin name.
- Latin name (accession).
- Common name.
- Common name (accession).

Whether sequences can be displayed with this information depends on their origin. Sequences that you have created yourself or imported might not include this information, and you will only be able to see them represented by their name. However, sequences downloaded from databases like GenBank will include this information. To change how sequences are displayed:

> **right-click any element or folder in the Navigation Area | Sequence Representation | select format**

This will only affect sequence elements, and the display of other types of elements, e.g. alignments, trees and external files, will be not be changed. If a sequence does not have this information, there will be no text next to the sequence icon.

Rename element

Renaming a folder or an element in the **Navigation Area** can be done in three different ways:

select the element | Edit in the Menu Bar | Rename

or **select the element | F2**

click the element once | wait one second | click the element again

When you can rename the element, you can see that the text is selected and you can move the cursor back and forth in the text. When the editing of the name has finished; press **Enter** or select another element in the **Navigation Area**. If you want to discard the changes instead, press the **Esc**-key.

For renaming annotations instead of folders or elements, see section 10.3.3.

3.1.7 Delete elements

Deleting a folder or an element can be done in two ways:

right-click the element | Delete (🗋)

or **select the element | press Delete key**

This will cause the element to be moved to the **Recycle Bin** (🗑) where it is kept until the recycle bin is emptied. This means that you can recover deleted elements later on.

For deleting annotations instead of folders or elements, see section 10.3.4.

Restore Deleted Elements

The elements in the **Recycle Bin** (🗑) can be restored by dragging the elements with the mouse into the folder where they used to be.

If you have deleted large amounts of data taking up very much disk space, you can free this disk space by emptying the **Recycle Bin** (🗑):

Edit in the Menu Bar | Empty Recycle Bin (🗑)

Note! This cannot be undone, and you will therefore not be able to recover the data present in the recycle bin when it was emptied.

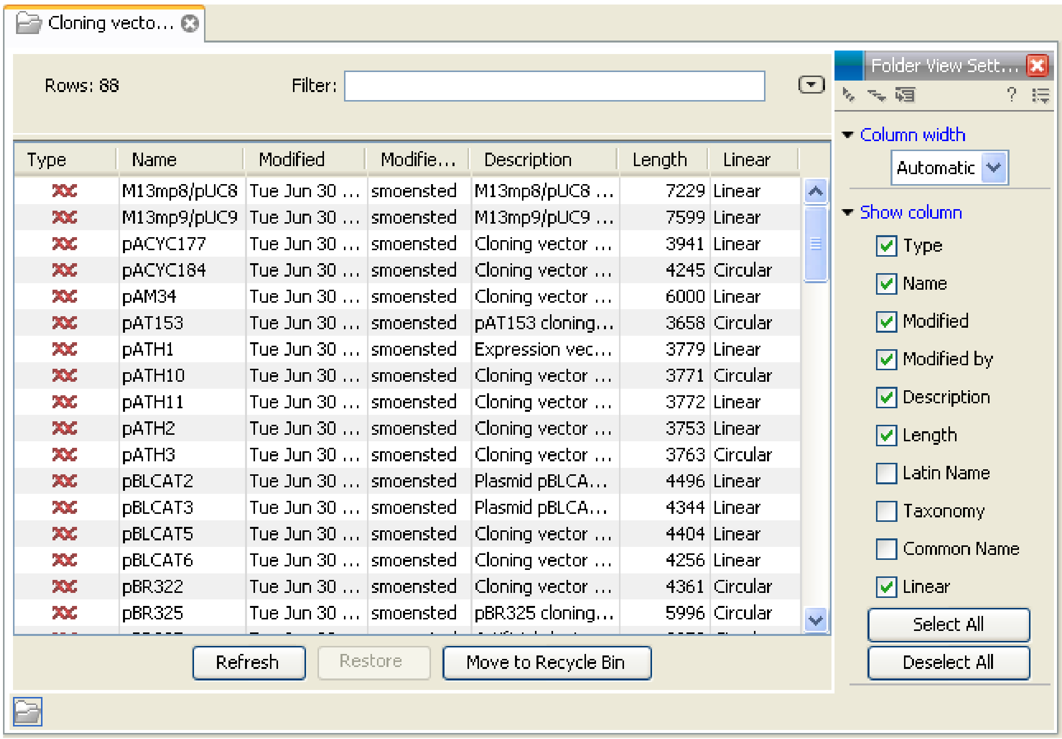
3.1.8 Show folder elements in a table

A location or a folder might contain large amounts of elements. It is possible to view their elements in the **View Area**:

select a folder or location | Show (🗎) in the Toolbar | Contents (📁)

An example is shown in figure 3.6.

When the elements are shown in the view, they can be sorted by clicking the heading of each of the columns. You can further refine the sorting by pressing Ctrl (⌘ on Mac) while clicking the heading of another column.



Rows: 88 Filter:

Type	Name	Modified	Modifie...	Description	Length	Linear
	M13mp8/pUC8	Tue Jun 30 ...	smoensted	M13mp8/pUC8 ...	7229	Linear
	M13mp9/pUC9	Tue Jun 30 ...	smoensted	M13mp9/pUC9 ...	7599	Linear
	pACYC177	Tue Jun 30 ...	smoensted	Cloning vector ...	3941	Linear
	pACYC184	Tue Jun 30 ...	smoensted	Cloning vector ...	4245	Circular
	pAM34	Tue Jun 30 ...	smoensted	Cloning vector ...	6000	Linear
	pAT153	Tue Jun 30 ...	smoensted	pAT153 cloning...	3658	Circular
	pATH1	Tue Jun 30 ...	smoensted	Expression vec...	3779	Linear
	pATH10	Tue Jun 30 ...	smoensted	Cloning vector ...	3771	Circular
	pATH11	Tue Jun 30 ...	smoensted	Cloning vector ...	3772	Linear
	pATH2	Tue Jun 30 ...	smoensted	Cloning vector ...	3753	Linear
	pATH3	Tue Jun 30 ...	smoensted	Cloning vector ...	3763	Circular
	pBLCAT2	Tue Jun 30 ...	smoensted	Plasmid pBLCA...	4496	Linear
	pBLCAT3	Tue Jun 30 ...	smoensted	Plasmid pBLCA...	4344	Linear
	pBLCAT5	Tue Jun 30 ...	smoensted	Cloning vector ...	4404	Linear
	pBLCAT6	Tue Jun 30 ...	smoensted	Cloning vector ...	4256	Linear
	pBR322	Tue Jun 30 ...	smoensted	Cloning vector ...	4361	Circular
	pBR325	Tue Jun 30 ...	smoensted	pBR325 cloning...	5996	Circular

Column width: Automatic

Show column

- ☑ Type
- ☑ Name
- ☑ Modified
- ☑ Modified by
- ☑ Description
- ☑ Length
- □ Latin Name
- □ Taxonomy
- □ Common Name
- ☑ Linear

Figure 3.6: *Viewing the elements in a folder.*

Sorting the elements in a view does not affect the ordering of the elements in the **Navigation Area**.

Note! The view only displays one "layer" at a time: the content of subfolders is not visible in this view. Also note that only sequences have the full span of information like organism etc.

Batch edit folder elements

You can select a number of elements in the table, right-click and choose **Edit** to batch edit the elements. In this way, you can change the e.g. the description or common name of several elements in one go.

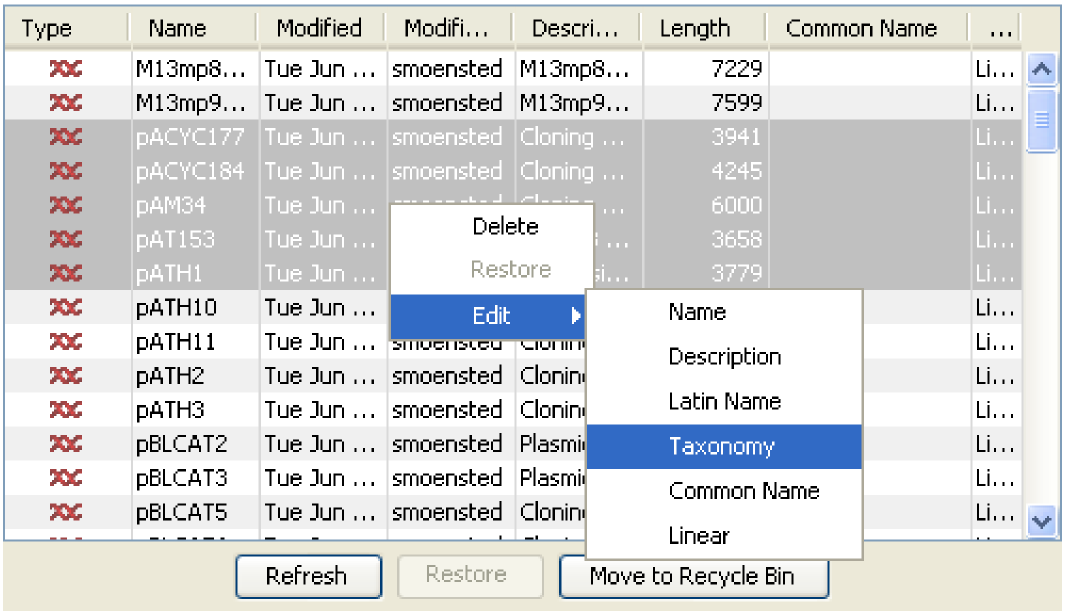
In figure 3.7 you can see an example where the common name of five sequence are renamed in one go. In this example, a dialog with a text field will be shown, letting you enter a new common name for these five sequences. **Note!** This information is directly saved and you cannot undo.

3.2 View Area

The **View Area** is the right-hand part of the screen, displaying your current work. The **View Area** may consist of one or more **Views**, represented by tabs at the top of the **View Area**.

This is illustrated in figure 3.8.

The tab concept is central to working with *CLC Genomics Workbench*, because several operations can be performed by dragging the tab of a view, and extended right-click menus can be activated from the tabs.



Type	Name	Modified	Modifi...	Descri...	Length	Common Name	...
	M13mp8...	Tue Jun ...	smoensted	M13mp8...	7229		Li...
	M13mp9...	Tue Jun ...	smoensted	M13mp9...	7599		Li...
	pACYC177	Tue Jun ...	smoensted	Cloning ...	3941		Li...
	pACYC184	Tue Jun ...	smoensted	Cloning ...	4245		Li...
	pAM34	Tue Jun ...		...	6000		Li...
	pAT153	Tue Jun ...		...	3658		Li...
	pATH1	Tue Jun ...		...	3779		Li...
	pATH10	Tue Jun ...					Li...
	pATH11	Tue Jun ...					Li...
	pATH2	Tue Jun ...	smoensted	Clonin			Li...
	pATH3	Tue Jun ...	smoensted	Clonin			Li...
	pBLCAT2	Tue Jun ...	smoensted	Plasmi			Li...
	pBLCAT3	Tue Jun ...	smoensted	Plasmi			Li...
	pBLCAT5	Tue Jun ...	smoensted	Clonin			Li...

Figure 3.7: *Changing the common name of five sequences.*

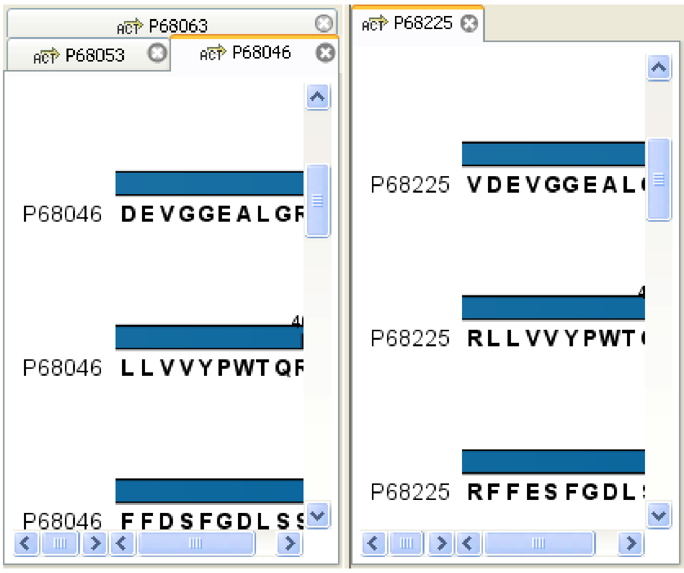


Figure 3.8: *A View Area can enclose several views, each view is indicated with a tab (see right view, which shows protein P68225). Furthermore, several views can be shown at the same time (in this example, four views are displayed).*

This chapter deals with the handling of views inside a **View Area**. Furthermore, it deals with rearranging the views.

Section 3.3 deals with the zooming and selecting functions.

3.2.1 Open view

Opening a view can be done in a number of ways:

double-click an element in the Navigation Area

or **select an element in the Navigation Area | File | Show | Select the desired way to view the element**

or **select an element in the Navigation Area | Ctrl + O (⌘ + B on Mac)**

Opening a view while another view is already open, will show the new view in front of the other view. The view that was already open can be brought to front by clicking its tab.

Note! If you right-click an open tab of any element, click **Show**, and then choose a different view of the same element, this new view is automatically opened in a split-view, allowing you to see both views.

See section 3.1.5 for instructions on how to open a view using drag and drop.

3.2.2 Show element in another view

Each element can be shown in different ways. A sequence, for example, can be shown as linear, circular, text etc.

In the following example, you want to see a sequence in a circular view. If the sequence is already open in a view, you can change the view to a circular view:

Click Show As Circular (○) at the lower left part of the view

The buttons used for switching views are shown in figure 3.9).

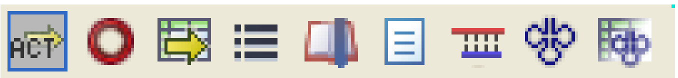


Figure 3.9: *The buttons shown at the bottom of a view of a nucleotide sequence. You can click the buttons to change the view to e.g. a circular view or a history view.*

If the sequence is already open in a linear view (ACT), and you wish to see both a circular and a linear view, you can split the views very easily:

Press Ctrl (⌘ on Mac) while you | Click Show As Circular (○) at the lower left part of the view

This will open a split view with a linear view at the bottom and a circular view at the top (see 10.5).

You can also show a circular view of a sequence without opening the sequence first:

Select the sequence in the Navigation Area | Show ( ) | As Circular (○)

3.2.3 Close views

When a view is closed, the **View Area** remains open as long as there is at least one open view.

A view is closed by:

right-click the tab of the View | Close

or **select the view | Ctrl + W**

or **hold down the Ctrl-button | Click the tab of the view while the button is pressed**

By right-clicking a tab, the following close options exist. See figure 3.10

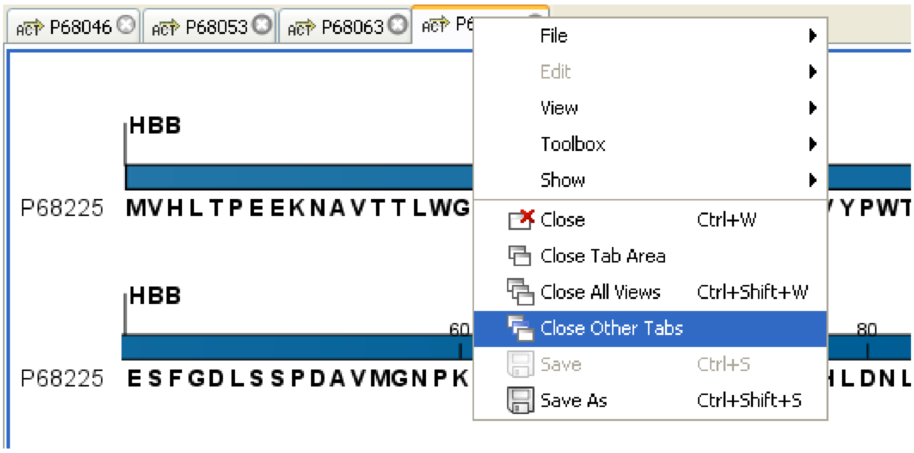


Figure 3.10: *By right-clicking a tab, several close options are available.*

- **Close.** See above.
- **Close Tab Area.** Closes all tabs in the tab area.
- **Close All Views.** Closes all tabs, in all tab areas. Leaves an empty workspace.
- **Close Other Tabs.** Closes all other tabs in the particular tab area.

3.2.4 Save changes in a view

When changes are made in a view, the text on the tab appears *bold and italic* (on Mac it is indicated by an * before the name of the tab). This indicates that the changes are not saved. The **Save** function may be activated in two ways:

Click the tab of the view you want to save | Save (🖫) in the toolbar.

or **Click the tab of the view you want to save | Ctrl + S (⌘ + S on Mac)**

If you close a view containing an element that has been changed since you opened it, you are asked if you want to save.

When saving a new view that has not been opened from the Navigation Area (e.g. when opening a sequence from a list of search hits), a save dialog appears (figure 3.11).

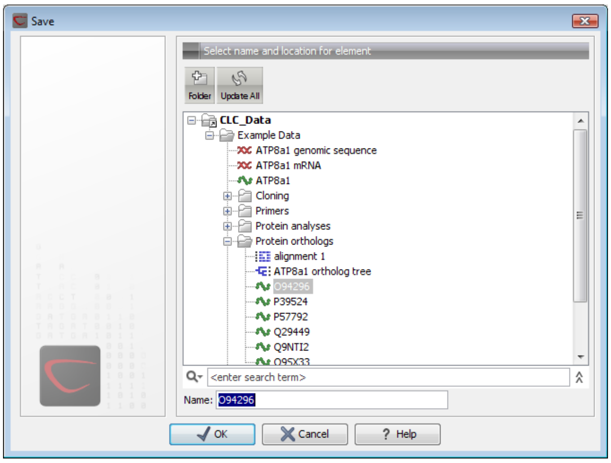


Figure 3.11: *Save dialog.*

In the dialog you select the folder in which you want to save the element.

After naming the element, press **OK**

3.2.5 Undo/Redo

If you make a change in a view, e.g. remove an annotation in a sequence or modify a tree, you can undo the action. In general, **Undo** applies to all changes you can make when right-clicking in a view. **Undo** is done by:

Click undo (↶) in the Toolbar

or **Edit | Undo (↶)**

or **Ctrl + Z**

If you want to undo several actions, just repeat the steps above. To reverse the undo action:

Click the redo icon in the Toolbar

or **Edit | Redo (↷)**

or **Ctrl + Y**

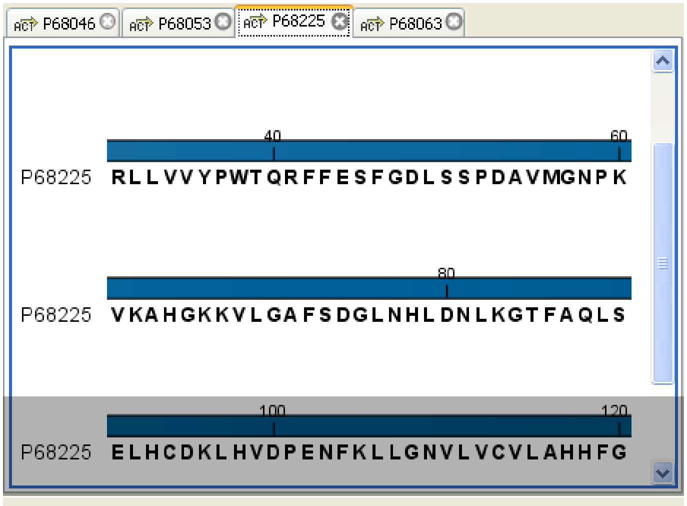
Note! Actions in the **Navigation Area**, e.g. renaming and moving elements, cannot be undone. However, you can restore deleted elements (see section 3.1.7).

You can set the number of possible undo actions in the Preferences dialog (see section 5).

3.2.6 Arrange views in View Area

Views are arranged in the **View Area** by their tabs. The order of the **views** can be changed using drag and drop. E.g. drag the tab of one view onto the tab of a another. The tab of the first view is now placed at the right side of the other tab.

If a tab is dragged into a view, an area of the view is made gray (see fig. 3.12) illustrating that the view will be placed in this part of the **View Area**.



```
P68225  RLLVVYPWTQRFFESFGDLSSPDAVMGNPK
P68225  VKAHGKKVLGAFSDGLNHLDNLKGTFAQLS
P68225  ELHCDKLHVDPENFKLLGNVLVCVLAHHFG
```

Figure 3.12: *When dragging a view, a gray area indicates where the view will be shown.*

The results of this action is illustrated in figure 3.13.

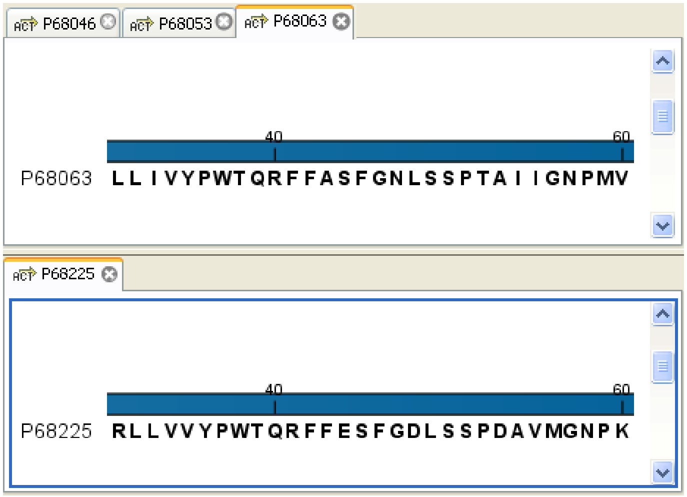

P68046
P68053
P68063
40
60
P68063 LLIVYPWTQRFFASFGNLSSPTAIIGNPMV
P68225
40
60
P68225 RLLVVYPWTQRFFESFGDLSSPDAVMGNPK

Figure 3.13: *A horizontal split-screen. The two views split the View Area.*

You can also split a **View Area** horizontally or vertically using the menus.

Splitting horisontally may be done this way:

> **right-click a tab of the view** | **View** | **Split Horizontally (⊟)**

This action opens the chosen view below the existing view. (See figure 3.14). When the split is made vertically, the new view opens to the right of the existing view.

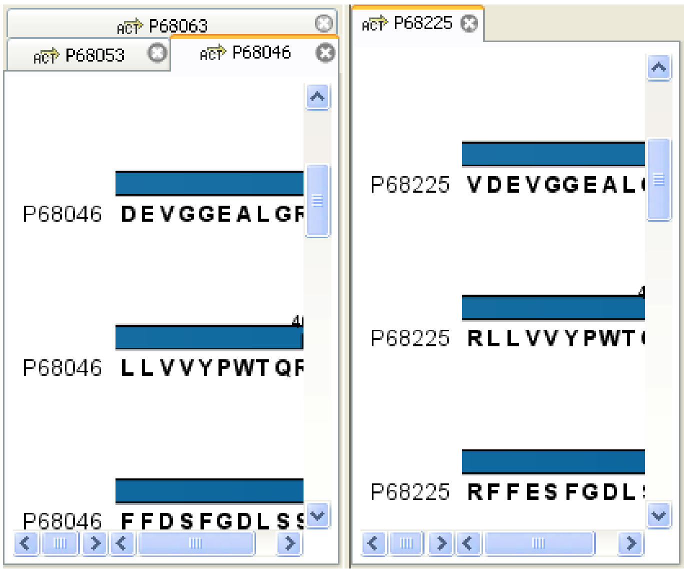
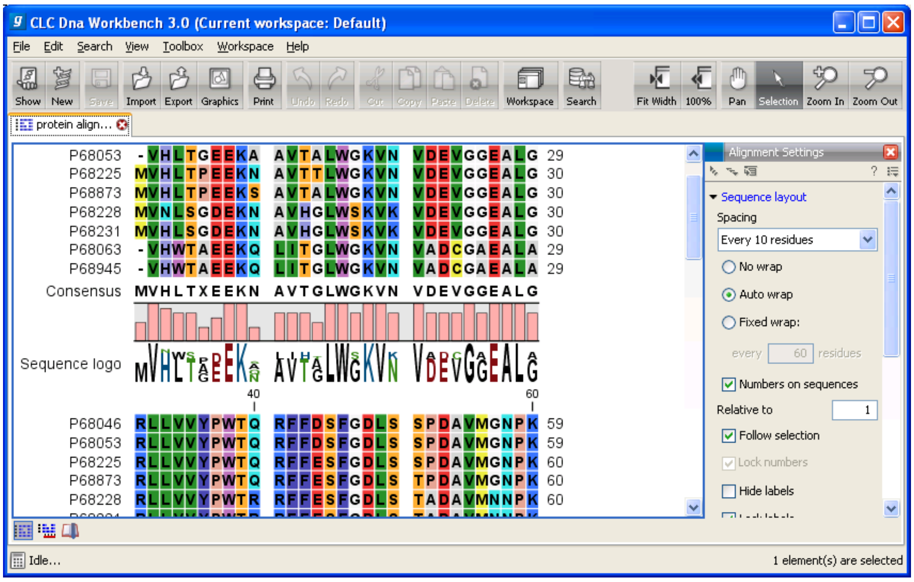


Figure 3.14: *A vertical split-screen.*

Splitting the **View Area** can be undone by dragging e.g. the tab of the bottom view to the tab of the top view. This is marked by a gray area on the top of the view.

Maximize/Restore size of view

The **Maximize/Restore View** function allows you to see a view in maximized mode, meaning a mode where no other **views** nor the **Navigation Area** is shown.


CLC Dna Workbench 3.0 (Current workspace: Default)
File Edit Search View Toolbox Workspace Help
Show New Save Import Export Graphics Print Undo Redo Cut Copy Paste Delete Workspace Search Fit Width 100% Pan Selection Zoom In Zoom Out
protein align...
Alignment Settings
Sequence layout
Idle...

```
    P68053  -VHLTGEEKA  AVTALWGKVN  VDEVGGEALG 29
    P68225  MVHLTPEEKN  AVTTLWGKVN  VDEVGGEALG 30
    P68873  MVHLTPEEKS  AVTALWGKVN  VDEVGGEALG 30
    P68228  MVNLSGDEKN  AVHGLWSKVK  VDEVGGEALG 30
    P68231  MVHLSGDEKN  AVHGLWSKVK  VDEVGGEALG 30
    P68063  -VHWTAEEKQ  LITGLWGKVN  VADCGAEALA 29
    P68945  -VHWTAEEKQ  LITGLWGKVN  VADCGAEALA 29
 Consensus  MVHLTXEEKN  AVTGLWGKVN  VDEVGGEALG

Sequence logo

                     40                   60
                      |                    |
    P68046  RLLVVYPWTQ  RFFDSFGDLS  SPDAVMGNPK 59
    P68053  RLLVVYPWTQ  RFFDSFGDLS  SPDAVMGNPK 59
    P68225  RLLVVYPWTQ  RFFESFGDLS  SPDAVMGNPK 60
    P68873  RLLVVYPWTQ  RFFESFGDLS  TPDAVMGNPK 60
    P68228  RLLVVYPWTR  RFFESFGDLS  TADAVMNNPK 60
```

Spacing: Every 10 residues

○ No wrap

● Auto wrap

○ Fixed wrap:

every 60 residues

☑ Numbers on sequences

Relative to 1

☑ Follow selection

☑ Lock numbers

□ Hide labels

1 element(s) are selected

Figure 3.15: *A maximized view. The function hides the Navigation Area and the Toolbox.*

Maximizing a view can be done in the following ways:

select view | Ctrl + M

or **select view | View | Maximize/restore View (□)**

or **select view | right-click the tab | View | Maximize/restore View (□)**

or **double-click the tab of view**

The following restores the size of the view:

Ctrl + M

or **View | Maximize/restore View (□)**

or **double-click title of view**

3.2.7 Side Panel

The **Side Panel** allows you to change the way the contents of a view are displayed. The options in the **Side Panel** depend on the kind of data in the view, and they are described in the relevant sections about sequences, alignments, trees etc.

Side Panel are activated in this way:

select the view | Ctrl + U (⌘ + U on Mac)

or **right-click the tab of the view | View | Show/Hide Side Panel ( )**

Note! Changes made to the **Side Panel** will not be saved when you save the view. See how to save the changes in the **Side Panel** in chapter 5 .

The **Side Panel** consists of a number of groups of preferences (depending on the kind of data being viewed), which can be expanded and collapsed by clicking the header of the group. You can also expand or collapse all the groups by clicking the icons ( )/ ( ) at the top.

3.3 Zoom and selection in View Area

The mode toolbar items in the right side of the **Toolbar** apply to the function of the mouse pointer. When e.g. **Zoom Out** is selected, you zoom out each time you click in a view where zooming is relevant (texts, tables and lists cannot be zoomed). The chosen mode is active until another mode toolbar item is selected. (**Fit Width** and **Zoom to 100%** do not apply to the mouse pointer.)

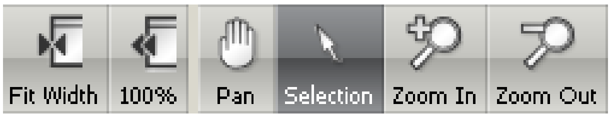


Figure 3.16: *The mode toolbar items.*

3.3.1 Zoom In

There are four ways of **Zooming In**:

Click Zoom In ( ) in the toolbar | click the location in the view that you want to. zoom in on

or **Click Zoom In ( ) in the toolbar | click-and-drag a box around a part of the view | the view now zooms in on the part you selected**

or **Press '+' on your keyboard**

The last option for zooming in is only available if you have a mouse with a scroll wheel:

or **Press and hold Ctrl (⌘ on Mac) | Move the scroll wheel on your mouse forward**

When you choose the Zoom In mode, the mouse pointer changes to a magnifying glass to reflect the mouse mode.

Note! You might have to click in the view before you can use the keyboard or the scroll wheel to zoom.

If you press the **Shift** button on your keyboard while clicking in a **View**, the zoom function is reversed. Hence, clicking on a sequence in this way while the **Zoom In** mode toolbar item is selected, zooms out instead of zooming in.

3.3.2 Zoom Out

It is possible to zoom out, step by step, on a sequence:

Click Zoom Out ( ) in the toolbar | click in the view until you reach a satisfying. zoomlevel

or **Press '-' on your keyboard**

The last option for zooming out is only available if you have a mouse with a scroll wheel:

or **Press and hold Ctrl (⌘ on Mac) | Move the scroll wheel on your mouse backwards**

When you choose the Zoom Out mode, the mouse pointer changes to a magnifying glass to reflect the mouse mode.

Note! You might have to click in the view before you can use the keyboard or the scroll wheel to zoom.

If you want to get a quick overview of a sequence or a tree, use the **Fit Width** function instead of the **Zoom Out** function.

If you press **Shift** while clicking in a **View**, the zoom function is reversed. Hence, clicking on a sequence in this way while the **Zoom Out** mode toolbar item is selected, zooms in instead of zooming out.

3.3.3 Fit Width

The **Fit Width** ( ) function adjusts the content of the **View** so that both ends of the sequence, alignment, or tree is visible in the **View** in question. (This function does not change the mode of the mouse pointer.)

3.3.4 Zoom to 100%

The **Zoom to 100%** ( ) function zooms the content of the **View** so that it is displayed with the highest degree of detail. (This function does not change the mode of the mouse pointer.)

3.3.5 Move

The Move mode allows you to drag the content of a **View**. E.g. if you are studying a sequence, you can click anywhere in the sequence and hold the mouse button. By moving the mouse you move the sequence in the **View**.

3.3.6 Selection

The Selection mode ( ) is used for selecting in a **View** (selecting a part of a sequence, selecting nodes in a tree etc.). It is also used for moving e.g. branches in a tree or sequences in an alignment.

When you make a selection on a sequence or in an alignment, the location is shown in the bottom right corner of the screen. E.g. ’23ˆ24’ means that the selection is between two residues. ’23’ means that the residue at position 23 is selected, and finally ’23..25’ means that 23, 24 and 25 are selected. By holding ctrl / ⌘ you can make multiple selections.

3.4 Toolbox and Status Bar

The **Toolbox** is placed in the left side of the user interface of *CLC Genomics Workbench* below the **Navigation Area**.

The **Toolbox** shows a **Processes tab** and a **Toolbox tab**.

3.4.1 Processes

By clicking the **Processes** tab, the **Toolbox** displays previous and running processes, e.g. an NCBI search or a calculation of an alignment. The running processes can be stopped, paused, and resumed by clicking the small icon (▾) next to the process (see figure 3.17).

Running and paused processes are not deleted.

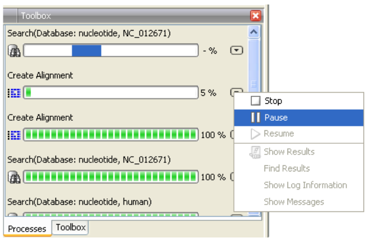


Figure 3.17: *A database search and an alignment calculation are running. Clicking the small icon next to the process allow you to stop, pause and resume processes.*

Besides the options to stop, pause and resume processes, there are some extra options for *a selected number* of the tools running from the Toolbox:

- **Show results**. If you have chosen to save the results (see section 9.2), you will be able to open the results directly from the process by clicking this option.
- **Find results**. If you have chosen to save the results (see section 9.2), you will be able to high-light the results in the Navigation Area.
- **Show Log Information**. This will display a log file showing progress of the process. The log file can also be shown by clicking **Show Log** in the "handle results" dialog where you choose between saving and opening the results.
- **Show Messages**. Some analyses will give you a message when processing your data. The messages are the black dialogs shown in the lower left corner of the Workbench that disappear after a few seconds. You can reiterate the messages that have been shown by clicking this option.

The terminated processes can be removed by:

View | Remove Terminated Processes (✖)

If you close the program while there are running processes, a dialog will ask if you are sure that you want to close the program. Closing the program will stop the process, and it cannot be restarted when you open the program again.

3.4.2 Toolbox

The content of the **Toolbox** tab in the **Toolbox** corresponds to **Toolbox** in the **Menu Bar**.

The **Toolbox** can be hidden, so that the **Navigation Area** is enlarged and thereby displays more elements:

View | Show/Hide Toolbox

The tools in the toolbox can be accessed by double-clicking or by dragging elements from the **Navigation Area** to an item in the **Toolbox**.

3.4.3 Status Bar

As can be seen from figure 3.1, the **Status Bar** is located at the bottom of the window. In the left side of the bar is an indication of whether the computer is making calculations or whether it is idle. The right side of the **Status Bar** indicates the range of the selection of a sequence. (See chapter 3.3.6 for more about the Selection mode button.)

3.5 Workspace

If you are working on a project and have arranged the views for this project, you can save this arrangement using **Workspaces**. A Workspace remembers the way you have arranged the views, and you can switch between different workspaces.

The **Navigation Area** always contains the same data across **Workspaces**. It is, however, possible to open different folders in the different **Workspaces**. Consequently, the program allows you to display different clusters of the data in separate **Workspaces**.

All **Workspaces** are automatically saved when closing down *CLC Genomics Workbench*. The next time you run the program, the **Workspaces** are reopened exactly as you left them.

Note! It is not possible to run more than one version of *CLC Genomics Workbench* at a time. Use two or more **Workspaces** instead.

3.5.1 Create Workspace

When working with large amounts of data, it might be a good idea to split the work into two or more **Workspaces**. As default the *CLC Genomics Workbench* opens one **Workspace**. Additional **Workspaces** are created in the following way:

Workspace in the Menu Bar) | Create Workspace | enter name of Workspace | OK

When the new **Workspace** is created, the heading of the program frame displays the name of the new **Workspace**. Initially, the selected elements in the **Navigation Area** is collapsed and the **View Area** is empty and ready to work with. (See figure 3.18).

3.5.2 Select Workspace

When there is more than one **Workspace** in the *CLC Genomics Workbench*, there are two ways to switch between them:

Workspace ( ) in the Toolbar | Select the Workspace to activate

or **Workspace in the Menu Bar | Select Workspace ( ) | choose which Workspace to activate | OK**

The name of the selected **Workspace** is shown after "*CLC Genomics Workbench*" at the top left corner of the main window, in figure 3.18 it says: (default).

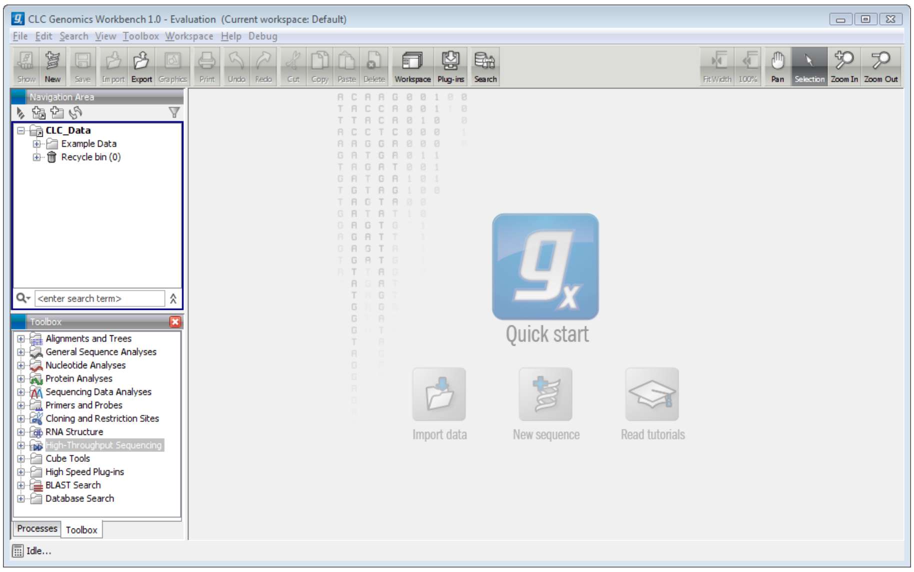

CLC Genomics Workbench 1.0 - Evaluation (Current workspace: Default)
File Edit Search View Toolbox Workspace Help Debug
Show New Save Import Export Graphics Print Undo Redo Cut Copy Paste Delete Workspace Plug-ins Search
Fit Width 100% Pan Selection Zoom In Zoom Out
Navigation Area
CLC_Data
Example Data
Recycle bin (0)
<enter search term>
Toolbox
Alignments and Trees
General Sequence Analyses
Nucleotide Analyses
Protein Analyses
Sequencing Data Analyses
Primers and Probes
Cloning and Restriction Sites
RNA Structure
High-Throughput Sequencing
Cube Tools
High Speed Plug-ins
BLAST Search
Database Search
Processes
Toolbox
Idle...
Quick start
Import data
New sequence
Read tutorials

Figure 3.18: *An empty Workspace.*

3.5.3 Delete Workspace

Deleting a **Workspace** can be done in the following way:

> **Workspace in the Menu Bar | Delete Workspace | choose which Workspace to delete | OK**

Note! Be careful to select the right **Workspace** when deleting. The delete action cannot be undone. (However, no data is lost, because a workspace is only a representation of data.)

It is not possible to delete the default workspace.

3.6 List of shortcuts

The keyboard shortcuts in *CLC Genomics Workbench* are listed below.

Action	**Windows/Linux**	**Mac OS X**
Adjust selection	Shift + arrow keys	Shift + arrow keys
Change between tabs[1]	Ctrl + tab	Ctrl + Page Up/Down
Close	Ctrl + W	⌘ + W
Close all views	Ctrl + Shift + W	⌘ + Shift + W
Copy	Ctrl + C	⌘ + C
Cut	Ctrl + X	⌘ + X
Delete	Delete	Delete or ⌘ + Backspace
Exit	Alt + F4	⌘ + Q
Export	Ctrl + E	⌘ + E
Export graphics	Ctrl + G	⌘ + G
Find Next Conflict	Space or .	Space or .
Find Previous Conflict	,	,
Help	F1	F1
Import	Ctrl + I	⌘ + I
Maximize/restore size of View	Ctrl + M	⌘ + M
Move gaps in alignment	Ctrl + arrow keys	⌘ + arrow keys
Navigate sequence views	arrow keys	arrow keys
New Folder	Ctrl + Shift + N	⌘ + Shift + N
New Sequence	Ctrl + N	⌘ + N
View	Ctrl + O	⌘ + O
Paste	Ctrl + V	⌘ + V
Print	Ctrl + P	⌘ + P
Redo	Ctrl + Y	⌘ + Y
Rename	F2	F2
Save	Ctrl + S	⌘ + S
Search local data	Ctrl + F	⌘ + F
Search within a sequence	Ctrl + Shift + F	⌘ + Shift + F
Search NCBI	Ctrl + B	⌘ + B
Search UniProt	Ctrl + Shift + U	⌘ + Shift + U
Select All	Ctrl + A	⌘ + A
Selection Mode	Ctrl + 2	⌘ + 2
Show/hide Side Panel	Ctrl + U	⌘ + U
Sort folder	Ctrl + Shift + R	⌘ + Shift + R
Split Horizontally	Ctrl + T	⌘ + T
Split Vertically	Ctrl + J	⌘ + J
Undo	Ctrl + Z	⌘ + Z
User Preferences	Ctrl + K	⌘ + ;
Zoom In Mode	Ctrl + + (plus)	⌘ + 3
Zoom In (without clicking)	+ (plus)	+ (plus)
Zoom Out Mode	Ctrl + - (minus)	⌘ + 4
Zoom Out (without clicking)	- (minus)	- (minus)
Inverse zoom mode	press and hold Shift	press and hold Shift

Combinations of keys and mouse movements are listed below.

[1] On Linux changing tabs is accomplished using Ctrl + Page Up/Page Down

Action	Windows/Linux	Mac OS X	Mouse movement
Maximize View			Double-click the tab of the View
Restore View			Double-click the View title
Reverse zoom function	Shift	Shift	Click in view
Select multiple elements	Ctrl	⌘	Click elements
Select multiple elements	Shift	Shift	Click elements

"Elements" in this context refers to elements and folders in the **Navigation Area** selections on sequences, and rows in tables.

Chapter 4

Searching your data

Contents

There are two ways of doing text-based searches of your data, as described in this chapter:

- **Quick-search** directly from the search field in the **Navigation Area**.
- **Advanced search** which makes it easy to make more specific searches.

In most cases, quick-search will find what you need, but if you need to be more specific in your search criteria, the advanced search is preferable.

4.1 What kind of information can be searched?

Below is a list of the different kinds of information that you can search for (applies to both quick-search and the advanced search).

- **Name.** The name of a sequence, an alignment or any other kind of element. The name is what is displayed in the **Navigation Area** per default.
- **Length.** The length of the sequence.
- **Organism.** Sequences which contain information about organism can be searched. In this way, you could search for e.g. *Homo sapiens* sequences.
- **Database fields.** If your data is stored in a CLC Bioinformatics Database, you will be able to search for custom defined information. Read more in the database user manual.

Only the first item in the list, **Name**, is available for all kinds of data. The rest is only relevant for sequences.

If you wish to perform a search for sequence similarity, use Local BLAST (see section 12.2) instead.

4.2 Quick search

At the bottom of the **Navigation Area** there is a text field as shown in figure 4.1).

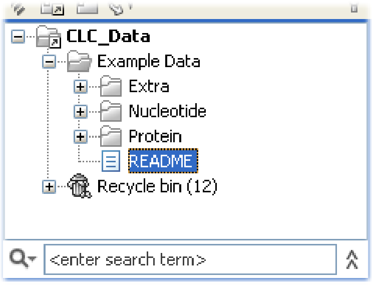


Figure 4.1: *Search simply by typing in the text field and press Enter.*

To search, simply enter a text to search for and press **Enter**.

4.2.1 Quick search results

To show the results, the search pane is expanded as shown in figure 4.2).

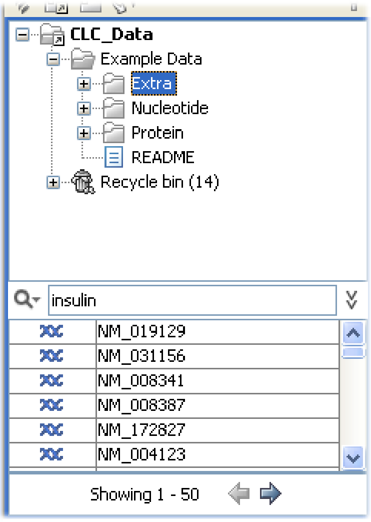
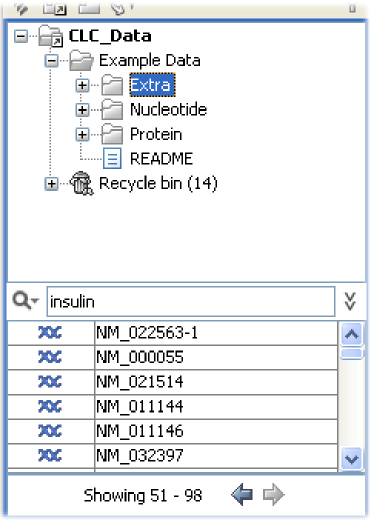


Figure 4.2: *Search results.*

If there are many hits, only the 50 first hits are immediately shown. At the bottom of the pane you can click **Next** (➡) to see the next 50 hits (see figure 4.3).

If a search gives no hits, you will be asked if you wish to search for matches that start with your search term. If you accept this, an asterisk (*) will be appended to the search term.

Pressing the Alt key while you click a search result will high-light the search hit in its folder in the **Navigation Area**.



CLC_Data
Example Data
Extra
Nucleotide
Protein
README
Recycle bin (14)

insulin

NM_022563-1
NM_000055
NM_021514
NM_011144
NM_011146
NM_032397

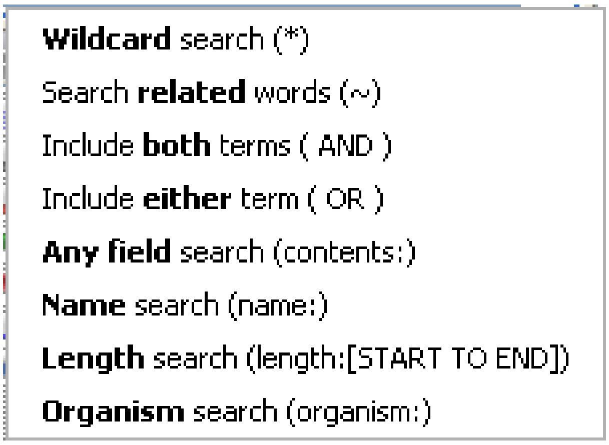
Showing 51 - 98

Figure 4.3: *Page two of the search results.*

In the preferences (see 5), you can specify the number of hits to be shown.

4.2.2 Special search expressions

When you write a search term in the search field, you can get help to write a more advanced search expression by pressing **Shift+F1**. This will reveal a list of guides as shown in figure 4.4.



Wildcard search (*)
Search **related** words (~)
Include **both** terms (AND)
Include **either** term (OR)
Any field search (contents:)
Name search (name:)
Length search (length:[START TO END])
Organism search (organism:)

Figure 4.4: *Guides to help create advanced search expressions.*

You can select any of the guides (using mouse or keyboard arrows), and start typing. If you e.g. wish to search for sequences named BRCA1, select "Name search (name:)", and type "BRCA1". Your search expression will now look like this: "name:BRCA1".

The guides available are these:

- **Wildcard search (*)**. Appending an asterisk * to the search term will find matches starting with the term. E.g. searching for "brca*" will find both *brca1* and *brca2*.
- **Search related words ( )**. If you don't know the exact spelling of a word, you can append a question mark to the search term. E.g. "brac1*" will find sequences with a *brca1* gene.

- **Include both terms (AND)**. If you write two search terms, you can define if your results have to match both search terms by combining them with AND. E.g. search for "brca1 AND human" will find sequences where *both* terms are present.
- **Include either term (OR)**. If you write two search terms, you can define that your results have to match either of the search terms by combining them with OR. E.g. search for "brca1 OR brca2" will find sequences where *either* of the terms is present.
- **Name search (name:)**. Search only the name of element.
- **Organism search (organism:)**. For sequences, you can specify the organism to search for. This will look in the "Latin name" field which is seen in the **Sequence Info** view (see section 10.4).
- **Length search (length:[START TO END])**. Search for sequences of a specific length. E.g. search for sequences between 1000 and 2000 residues: "length:1000 TO 2000".

If you do not use this special syntax, you will automatically search for both name, description, organism, etc., and search terms will be combined as if you had put OR between them.

4.2.3 Quick search history

You can access the 10 most recent searches by clicking the icon (🔍▾) next to the search field (see figure 4.5).

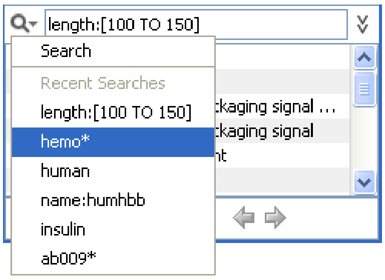


Figure 4.5: *Recent searches.*

Clicking one of the recent searches will conduct the search again.

4.3 Advanced search

As a supplement to the **Quick search** described in the previous section you can use the more advanced search:

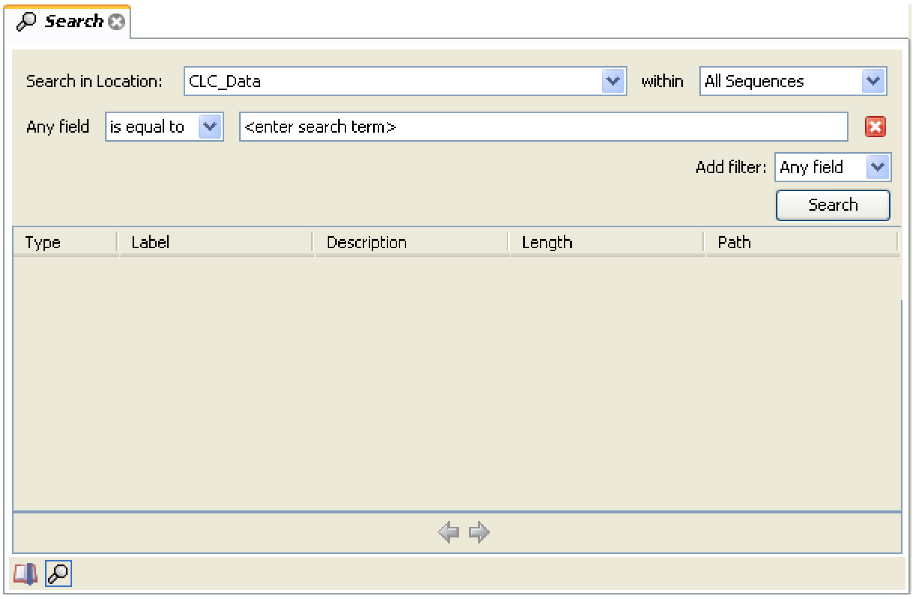
Search | Local Search (🔍)

or **Ctrl + F (⌘ + F on Mac)**

This will open the search view as shown in figure 4.6

The first thing you can choose is which location should be searched. All the active locations are shown in this list. You can also choose to search all locations. Read more about locations in section 3.1.1.

Furthermore, you can specify what kind of elements should be searched:



Search in Location: CLC_Data within All Sequences

Any field is equal to <enter search term>

Add filter: Any field

Figure 4.6: *Advanced search.*

- All sequences
- Nucleotide sequences
- Protein sequences
- All data

When searching for sequences, you will also get alignments, sequence lists etc as result, if they contain a sequence which match the search criteria.

Below are the search criteria. First, select a relevant search filter in the **Add filter:** list. For sequences you can search for

- Name
- Length
- Organism

See section 4.2.2 for more information on individual search terms.

For all other data, you can only search for name.

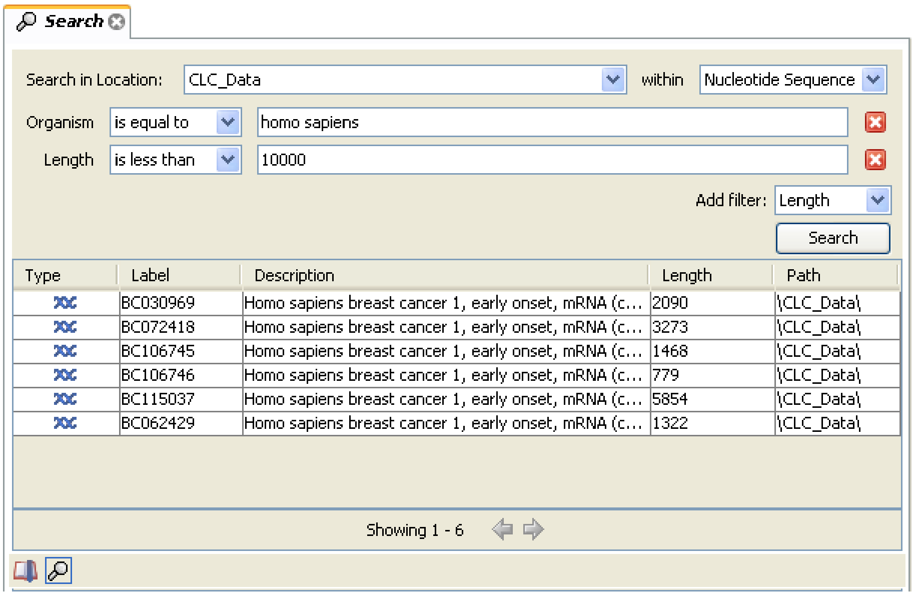
If you use **Any field**, it will search all of the above plus the following:

- Description
- Keywords
- Common name
- Taxonomy name

To see this information for a sequence, switch to the **Element Info** () view (see section 10.4).

For each search line, you can choose if you want the exact term by selecting "is equal to" or if you only enter the start of the term you wish to find (select "begins with").

An example is shown in figure 4.7.



Search in Location: CLC_Data within Nucleotide Sequence

Organism is equal to homo sapiens

Length is less than 10000

Add filter: Length

Type	Label	Description	Length	Path
	BC030969	Homo sapiens breast cancer 1, early onset, mRNA (c...	2090	\CLC_Data\
	BC072418	Homo sapiens breast cancer 1, early onset, mRNA (c...	3273	\CLC_Data\
	BC106745	Homo sapiens breast cancer 1, early onset, mRNA (c...	1468	\CLC_Data\
	BC106746	Homo sapiens breast cancer 1, early onset, mRNA (c...	779	\CLC_Data\
	BC115037	Homo sapiens breast cancer 1, early onset, mRNA (c...	5854	\CLC_Data\
	BC062429	Homo sapiens breast cancer 1, early onset, mRNA (c...	1322	\CLC_Data\

Showing 1 - 6

Figure 4.7: *Searching for human sequences shorter than 10,000 nucleotides.*

This example will find human nucleotide sequences (organism is *Homo sapiens*), and it will only find sequences shorter than 10,000 nucleotides.

Note that a search can be saved () for later use. You do not save the search results - only the search parameters. This means that you can easily conduct the same search later on when your data has changed.

4.4 Search index

This section has a technical focus and is not relevant if your search works fine.

However, if you experience problems with your search results: if you do not get the hits you expect, it might be because of an index error.

The *CLC Genomics Workbench* automatically maintains an index of all data in all locations in the **Navigation Area**. If this index becomes out of sync with the data, you will experience problems with strange results. In this case, you can rebuild the index:

> **Right-click the relevant location** | **Location** | **Rebuild Index**

This will take a while depending on the size of your data. At any time, the process can be stopped in the process area, see section 3.4.1.

Chapter 5

User preferences and settings

Contents

The first three sections in this chapter deal with the general preferences that can be set for *CLC Genomics Workbench* using the **Preferences** dialog. The next section explains how the settings in the **Side Panel** can be saved and applied to other views. Finally, you can learn how to import and export the preferences.

The **Preferences** dialog offers opportunities for changing the default settings for different features of the program.

The **Preferences** dialog is opened in one of the following ways and can be seen in figure 5.1:

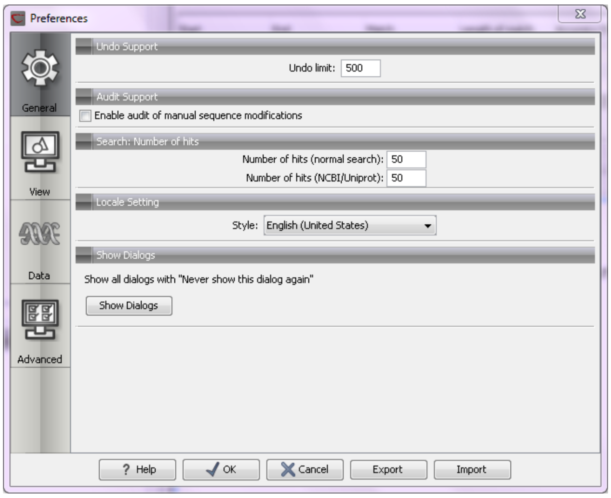
Edit | **Preferences** ( )

or **Ctrl + K (⌘ + ; on Mac)**

5.1 General preferences

The **General** preferences include:

- **Undo Limit.** As default the undo limit is set to 500. By writing a higher number in this field, more actions can be undone. Undo applies to all changes made on sequences, alignments or trees. See section 3.2.5 for more on this topic.



Undo Support

Undo limit: 500

Audit Support

□ Enable audit of manual sequence modifications

Search: Number of hits

Number of hits (normal search): 50

Number of hits (NCBI/Uniprot): 50

Locale Setting

Style: English (United States)

Show Dialogs

Show all dialogs with "Never show this dialog again"

Show Dialogs

Figure 5.1: *Preferences include General preferences, View preferences, Colors preferences, and Advanced settings.*

- **Audit Support.** If this option is checked, all manual editing of sequences will be marked with an annotation on the sequence (see figure 5.2). Placing the mouse on the annotation will reveal additional details about the change made to the sequence (see figure 5.3). Note that no matter whether **Audit Support** is checked or not, all changes are also recorded in the **History** () (see section 8).
- **Number of hits.** The number of hits shown in *CLC Genomics Workbench*, when e.g. searching NCBI. (The sequences shown in the program are not downloaded, until they are opened or dragged/saved into the Navigation Area.
- **Locale Setting.** Specify which country you are located in. This determines how punctation is used in numbers all over the program.
- **Show Dialogs.** A lot of information dialogs have a checkbox: "Never show this dialog again". When you see a dialog and check this box in the dialog, the dialog will not be shown again. If you regret and wish to have the dialog displayed again, click the button in the General Preferences: **Show Dialogs**. Then all the dialogs will be shown again.

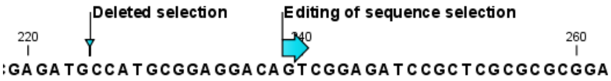
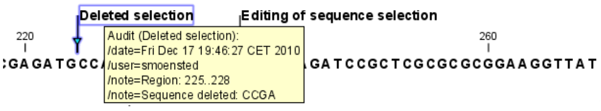

Deleted selection
Editing of sequence selection
220
240
260
GAGATGCCATGCGGAGGACAGTCGGAGATCCGCTCGCGCGCGGA

Figure 5.2: *Annotations added when the sequence is edited.*



Audit (Deleted selection):
/date=Fri Dec 17 19:46:27 CET 2010
/user=smoensted
/note=Region: 225..228
/note=Sequence deleted: CCGA

Figure 5.3: *Details of the editing.*

5.2 Default view preferences

There are five groups of default **View** settings:

1. **Toolbar**
2. **Side Panel Location**
3. **New View**
4. **View Format**
5. **User Defined View Settings**.

In general, these are default settings for the user interface.

The **Toolbar preferences** let you choose the size of the toolbar icons, and you can choose whether to display names below the icons.

The **Side Panel Location** setting lets you choose between **Dock in views** and **Float in window**. When docked in view, view preferences will be located in the right side of the view of e.g. an alignment. When floating in window, the side panel can be placed everywhere in your screen, also outside the workspace, e.g. on a different screen. See section 5.6 for more about floating side panels.

The **New view** setting allows you to choose whether the **View preferences** are to be shown automatically when opening a new view. If this option is not chosen, you can press (Ctrl + U (⌘ + U on Mac)) to see the preferences panels of an open view.

The **View Format** allows you to change the way the elements appear in the **Navigation Area**. The following text can be used to describe the element:

- Name (this is the default information to be shown).
- Accession (sequences downloaded from databases like GenBank have an accession number).
- Latin name.
- Latin name (accession).
- Common name.
- Common name (accession).

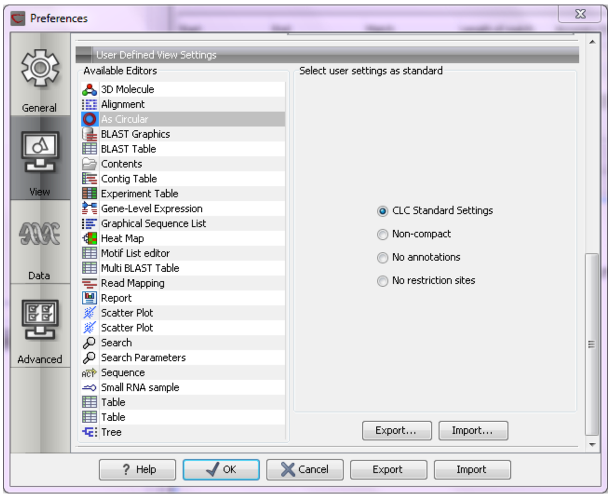
The **User Defined View Settings** gives you an overview of the different **Side Panel** settings that are saved for each view. See section 5.6 for more about how to create and save style sheets.

If there are other settings beside **CLC Standard Settings**, you can use this overview to choose which of the settings should be used per default when you open a view (see an example in figure 5.4).

In this example, the **CLC Standard Settings** is chosen as default.

5.2.1 Import and export Side Panel settings

If you have created a special set of settings in the **Side Panel** that you wish to share with other CLC users, you can export the settings in a file. The other user can then import the settings.



User Defined View Settings

Available Editors

- 3D Molecule
- Alignment
- As Circular
- BLAST Graphics
- BLAST Table
- Contents
- Contig Table
- Experiment Table
- Gene-Level Expression
- Graphical Sequence List
- Heat Map
- Motif List editor
- Multi BLAST Table
- Read Mapping
- Report
- Scatter Plot
- Scatter Plot
- Search
- Search Parameters
- Sequence
- Small RNA sample
- Table
- Table
- Tree

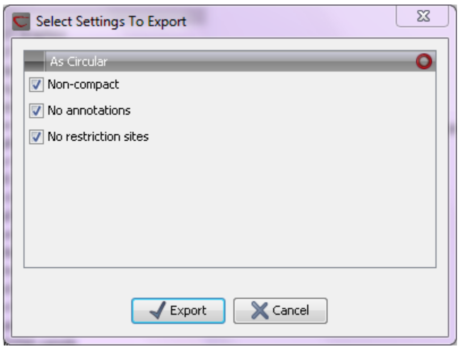
Select user settings as standard

- ● CLC Standard Settings
- ○ Non-compact
- ○ No annotations
- ○ No restriction sites

Figure 5.4: *Selecting the default view setting.*

To export the **Side Panel** settings, first select the views that you wish to export settings for. Use Ctrl+click (⌘ + click on Mac) or Shift+click to select multiple views. Next click the **Export...**button. Note that there is also another export button at the very bottom of the dialog, but this will export the other settings of the **Preferences** dialog (see section 5.5).

A dialog will be shown (see figure 5.5) that allows you to select which of the settings you wish to export.



As Circular

- ☑ Non-compact
- ☑ No annotations
- ☑ No restriction sites

Figure 5.5: *Exporting all settings for circular views.*

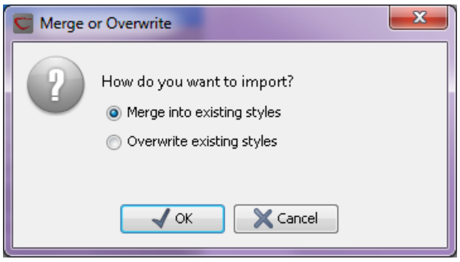
When multiple views are selected for export, all the view settings for the views will be shown in the dialog. Click **Export** and you will now be able to define a save folder and name for the exported file. The settings are saved in a file with a .vsf extension (View Settings File).

To import a **Side Panel** settings file, make sure you are at the bottom of the **View** panel of the **Preferences dialog**, and click the **Import...** button. Note that there is also another import button at the very bottom of the dialog, but this will import the other settings of the **Preferences** dialog (see section 5.5).

The dialog asks if you wish to overwrite existing **Side Panel** settings, or if you wish to merge the imported settings into the existing ones (see figure 5.6).

Note! If you choose to overwrite the existing settings, you will loose all the **Side Panel** settings that you have previously saved.

To avoid confusion of the different import and export options, here is an overview:



Merge or Overwrite

How do you want to import?

● Merge into existing styles

○ Overwrite existing styles

Figure 5.6: *When you import settings, you are asked if you wish to overwrite existing settings or if you wish to merge the new settings into the old ones.*

- Import and export of **bioinformatics data** such as sequences, alignments etc. (described in section 7.1.1).
- **Graphics** export of the views which creates image files in various formats (described in section 7.3).
- Import and export of **Side Panel Settings** as described above.
- Import and export of all the **Preferences** except the Side Panel settings. This is described in the previous section.

5.3 Data preferences

The data preferences contain preferences related to interpretation of data, e.g. linker sequences:

- Linkers for importing 454 data (see section 19.1.1).
- Predefined primer additions for Gateway cloning (see section 21.2.1).
- Adapter sequences for trimming (see section 19.3.2).

5.4 Advanced preferences

The **Advanced** settings include the possibility to set up a proxy server. This is described in section 1.8.

5.4.1 Default data location

If you have more than one location in the **Navigation Area**, you can choose which location should be the default data location. The default location is used when you e.g. import a file without selecting a folder or element in the **Navigation Area** first. Then the imported element will be placed in the default location.

Note! The default location cannot be removed. You have to select another location as default first.

5.5 Export/import of preferences

The user preferences of the *CLC Genomics Workbench* can be exported to other users of the program, allowing other users to display data with the same preferences as yours. You can also use the export/import preferences function to backup your preferences.

To export preferences, open the **Preferences** dialog (Ctrl + K (⌘ + ; on Mac)) and do the following:

> **Export | Select the relevant preferences | Export | Choose location for the exported file | Enter name of file | Save**

Note! The format of exported preferences is .cpf. This notation must be submitted to the name of the exported file in order for the exported file to work.

Before exporting, you are asked about which of the different settings you want to include in the exported file. One of the items in the list is "User Defined View Settings". If you export this, only the information about which of the settings is the default setting for each view is exported. If you wish to export the **Side Panel Settings** themselves, see section 5.2.1.

The process of importing preferences is similar to exporting:

> **Press Ctrl + K (⌘ + ; on Mac) to open Preferences | Import | Browse to and select the .cpf file | Import and apply preferences**

5.5.1 The different options for export and importing

To avoid confusion of the different import and export options, here is an overview:

- Import and export of **bioinformatics data** such as sequences, alignments etc. (described in section 7.1.1).
- **Graphics** export of the views which creates image files in various formats (described in section 7.3).
- Import and export of **Side Panel Settings** as described in the next section.
- Import and export of all the **Preferences** except the Side Panel settings. This is described above.

5.6 View settings for the Side Panel

The **Side Panel** is shown to the right of all views that are opened in *CLC Genomics Workbench*. By using the settings in the **Side Panel** you can specify how the layout and contents of the view. Figure 5.7 is an example of the **Side Panel** of a sequence view.

By clicking the black triangles or the corresponding headings, the groups can be expanded or collapsed. An example is shown in figure 5.8 where the **Sequence layout** is expanded.

The content of the groups is described in the sections where the functionality is explained. E.g. **Sequence Layout** for sequences is described in chapter 10.1.1.

When you have adjusted a view of e.g. a sequence, your settings in the **Side Panel** can be saved. When you open other sequences, which you want to display in a similar way, the saved settings can be applied. The options for saving and applying are available in the top of the **Side Panel** (see figure 5.9).

To save and apply the saved settings, click (☰) seen in figure 5.9. This opens a menu, where the following options are available:

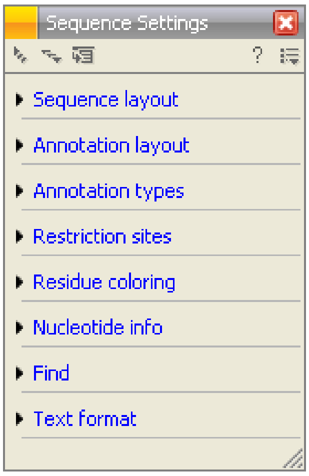
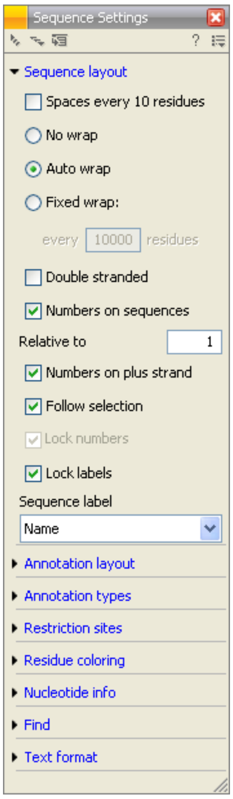


Figure 5.7: *The Side Panel of a sequence contains several groups: Sequence layout, Annotation types, Annotation layout, etc. Several of these groups are present in more views. E.g. Sequence layout is also in the Side Panel of alignment views.*


Sequence Settings

Sequence layout

□ Spaces every 10 residues

○ No wrap

● Auto wrap

○ Fixed wrap:

every 10000 residues

□ Double stranded

☑ Numbers on sequences

Relative to 1

☑ Numbers on plus strand

☑ Follow selection

☑ Lock numbers

☑ Lock labels

Sequence label: Name

Figure 5.8: *The Sequence layout is expanded.*

- **Save Settings.** This brings up a dialog as shown in figure 5.10 where you can enter a name for your settings. Furthermore, by clicking the checkbox **Always apply these settings**, you can choose to use these settings every time you open a new view of this type. If you wish to change which settings should be used per default, open the **Preferences** dialog (see section 5.2).
- **Delete Settings.** Opens a dialog to select which of the saved settings to delete.
- **Apply Saved Settings.** This is a submenu containing the settings that you have previously

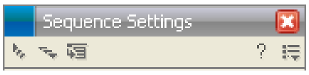
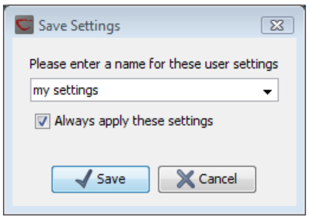


Figure 5.9: *At the top of the Side Panel you can: Expand all groups, Collapse all preferences, Dock/Undock preferences, Help, and Save/Restore preferences.*

saved. By clicking one of the settings, they will be applied to the current view. You will also see a number of pre-defined view settings in this submenu. They are meant to be examples of how to use the **Side Panel** and provide quick ways of adjusting the view to common usages. At the bottom of the list of settings you will see **CLC Standard Settings** which represent the way the program was set up, when you first launched it.


Save Settings

Please enter a name for these user settings

my settings

☑ Always apply these settings

Figure 5.10: *The save settings dialog.*

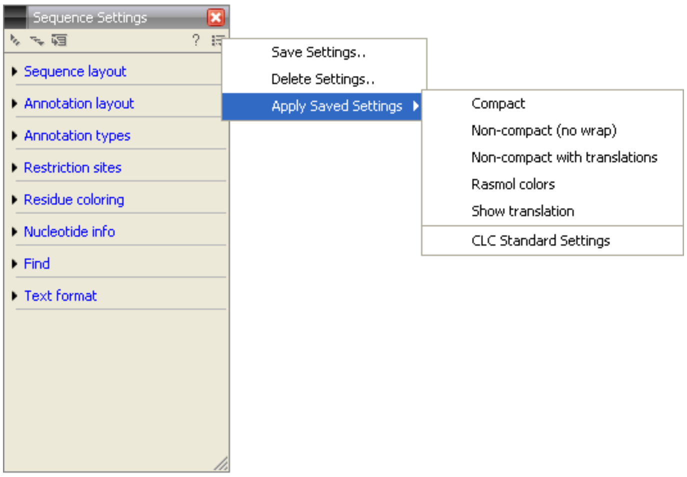


Figure 5.11: *Applying saved settings.*

The settings are specific to the type of view. Hence, when you save settings of a circular view, they will not be available if you open the sequence in a linear view.

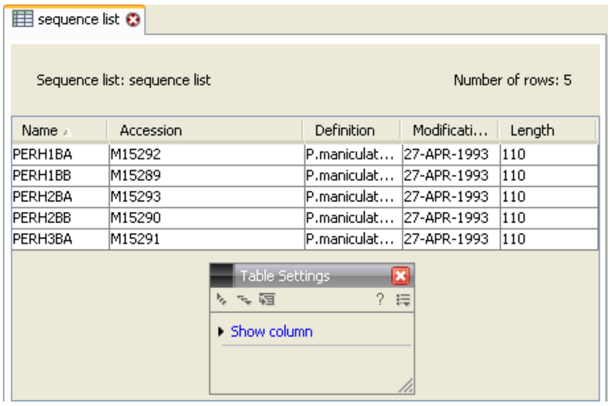
If you wish to export the settings that you have saved, this can be done in the **Preferences** dialog under the **View** tab (see section 5.2.1).

The remaining icons of figure 5.9 are used to; **Expand all groups**, **Collapse all groups**, and **Dock/Undock Side Panel**. **Dock/Undock Side Panel** is to make the **Side Panel** "floating" (see below).

5.6.1 Floating Side Panel

The Side Panel of the views can be placed in the right side of a view, or it can be floating (see figure 5.12).

By clicking the Dock icon (▤) the floating Side Panel reappear in the right side of the view. The size of the floating Side Panel can be adjusted by dragging the hatched area in the bottom right.



Sequence list: sequence list

Number of rows: 5

Name	Accession	Definition	Modificati...	Length
PERH1BA	M15292	P.maniculat...	27-APR-1993	110
PERH1BB	M15289	P.maniculat...	27-APR-1993	110
PERH2BA	M15293	P.maniculat...	27-APR-1993	110
PERH2BB	M15290	P.maniculat...	27-APR-1993	110
PERH3BA	M15291	P.maniculat...	27-APR-1993	110

Figure 5.12: *The floating Side Panel can be moved out of the way, e.g. to allow for a wider view of a table.*

Chapter 6

Printing

Contents

CLC Genomics Workbench offers different choices of printing the result of your work.

This chapter deals with printing directly from *CLC Genomics Workbench*. Another option for using the graphical output of your work, is to export graphics (see chapter 7.3) in a graphic format, and then import it into a document or a presentation.

All the kinds of data that you can view in the **View Area** can be printed. The *CLC Genomics Workbench* uses a WYSIWYG principle: What You See Is What You Get. This means that you should use the options in the Side Panel to change how your data, e.g. a sequence, looks on the screen. When you print it, it will look exactly the same way on print as on the screen.

For some of the views, the layout will be slightly changed in order to be printer-friendly.

It is not possible to print elements directly from the **Navigation Area**. They must first be opened in a view in order to be printed. To print the contents of a view:

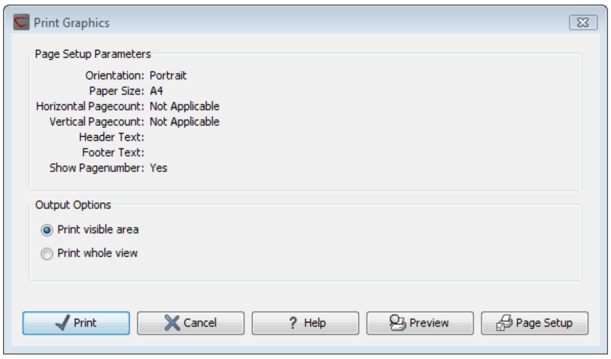
select relevant view | Print (🖨) in the toolbar

This will show a print dialog (see figure 6.1).

In this dialog, you can:

- Select which part of the view you want to print.
- Adjust **Page Setup**.
- See a print **Preview** window.

These three options are described in the three following sections.



Print Graphics

Page Setup Parameters

Orientation: Portrait
Paper Size: A4
Horizontal Pagecount: Not Applicable
Vertical Pagecount: Not Applicable
Header Text:
Footer Text:
Show Pagenumber: Yes

Output Options

● Print visible area
○ Print whole view

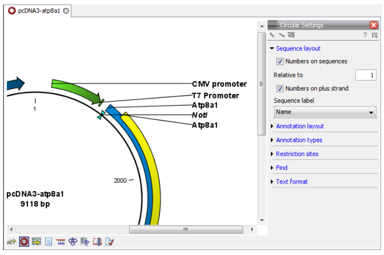
Figure 6.1: *The Print dialog.*

6.1 Selecting which part of the view to print

In the print dialog you can choose to:

- **Print visible area**, or
- **Print whole view**

These options are available for all views that can be zoomed in and out. In figure 6.2 is a view of a circular sequence which is zoomed in so that you can only see a part of it.



Circular Settings

Sequence layout
☑ Numbers on sequences
Relative to: 1
☑ Numbers on plus strand
Sequence label: Name

Figure 6.2: *A circular sequence as it looks on the screen.*

When selecting **Print visible area**, your print will reflect the part of the sequence that is *visible* in the view. The result from printing the view from figure 6.2 and choosing **Print visible area** can be seen in figure 6.3.

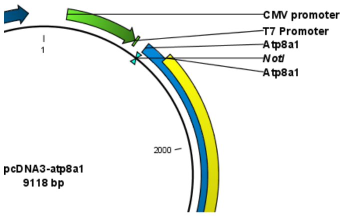


Figure 6.3: *A print of the sequence selecting Print visible area.*

On the other hand, if you select **Print whole view**, you will get a result that looks like figure 6.4. This means that you also print the part of the sequence which is not visible when you have zoomed in.

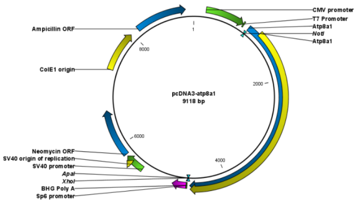
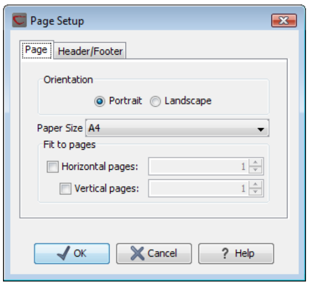


Figure 6.4: *A print of the sequence selecting Print whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*

6.2 Page setup

No matter whether you have chosen to print the visible area or the whole view, you can adjust page setup of the print. An example of this can be seen in figure 6.5



Orientation

● Portrait ○ Landscape

Paper Size A4

Fit to pages

□ Horizontal pages: 1

□ Vertical pages: 1

Figure 6.5: *Page Setup.*

In this dialog you can adjust both the setup of the pages and specify a header and a footer by clicking the tab at the top of the dialog.

You can modify the layout of the page using the following options:

- **Orientation**.
  - **Portrait**. Will print with the paper oriented vertically.
  - **Landscape**. Will print with the paper oriented horizontally.
- **Paper size**. Adjust the size to match the paper in your printer.
- **Fit to pages**. Can be used to control how the graphics should be split across pages (see figure 6.6 for an example).
  - **Horizontal pages**. If you set the value to e.g. 2, the printed content will be broken up horizontally and split across 2 pages. This is useful for sequences that are not wrapped
  - **Vertical pages**. If you set the value to e.g. 2, the printed content will be broken up vertically and split across 2 pages.

Note! It is a good idea to consider adjusting view settings (e.g. **Wrap** for sequences), in the **Side Panel** before printing. As explained in the beginning of this chapter, the printed material will look like the view on the screen, and therefore these settings should also be considered when adjusting **Page Setup**.

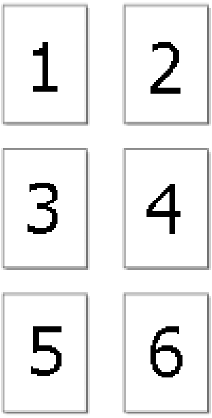


Figure 6.6: *An example where Fit to pages horizontally is set to 2, and Fit to pages vertically is set to 3.*

6.2.1 Header and footer

Click the **Header/Footer** tab to edit the header and footer text. By clicking in the text field for either **Custom header text** or **Custom footer text** you can access the auto formats for header/footer text in **Insert a caret position**. Click either **Date**, **View name**, or **User name** to include the auto format in the header/footer text.

Click **OK** when you have adjusted the **Page Setup**. The settings are saved so that you do not have to adjust them again next time you print. You can also change the **Page Setup** from the **File** menu.

6.3 Print preview

The preview is shown in figure 6.7.

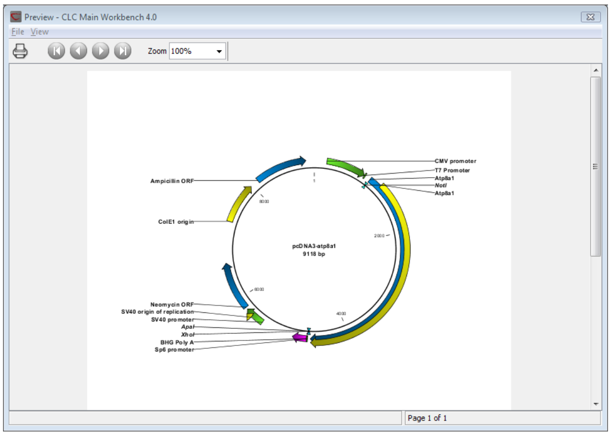


Figure 6.7: *Print preview.*

The **Print preview** window lets you see the layout of the pages that are printed. Use the arrows in the toolbar to navigate between the pages. Click Print (🖨) to show the print dialog, which lets you choose e.g. which pages to print.

The **Print preview** window is for preview only - the layout of the pages must be adjusted in the **Page setup**.

Chapter 7

Import/export of data and graphics

Contents

CLC Genomics Workbench handles a large number of different data formats. All data stored in the Workbench are available in the **Navigation Area**. The data of the **Navigation Area** can be divided into two groups. The data is either one of the different bioinformatic data formats, or it can be an 'external file'. Bioinformatic data formats are those formats which the program can work with, e.g. sequences, alignments and phylogenetic trees. External files are files or links which are stored in *CLC Genomics Workbench*, but are opened by other applications, e.g. pdf-files, Microsoft Word files, Open Office spreadsheet files, or links to programs and web-pages etc.

This chapter first deals with importing and exporting data in bioinformatic data formats and as external files. Next comes an explanation of how to export graph data points to a file, and how export graphics.

7.1 Bioinformatic data formats

The different bioinformatic data formats are imported in the same way, therefore, the following description of data import is an example which illustrates the general steps to be followed, regardless of which format you are handling.

7.1.1 Import of bioinformatic data

CLC Genomics Workbench has support for a wide range of bioinformatic data such as sequences, alignments etc. See a full list of the data formats in section J.1.

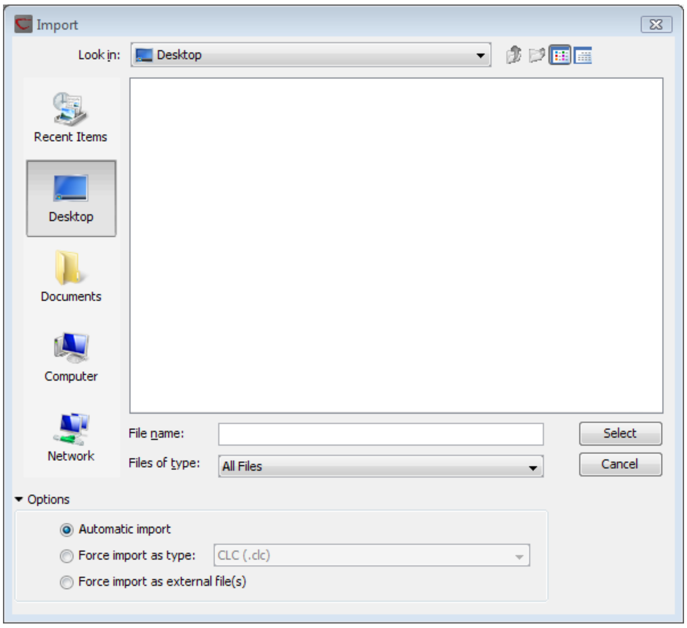
The *CLC Genomics Workbench* offers a lot of possibilities to handle bioinformatic data. Read the next sections to get information on how to import different file formats or to import data from a Vector NTI database.

Import using the import dialog

Before importing a file, you must decide where you want to import it, i.e. which location or folder. The imported file ends up in the location or folder you selected in the **Navigation Area**.

> **select location or folder | click Import ( ) in the Toolbar**

This will show a dialog similar to figure 7.1 (depending on which platform you use). You can change which kind of file types that should be shown by selecting a file format in the **Files of type** box.



Look in: Desktop

File name:

Files of type: All Files

● Automatic import

○ Force import as type: CLC (.clc)

○ Force import as external file(s)

Figure 7.1: *The import dialog.*

Next, select one or more files or folders to import and click **Select**.

The imported files are placed at the location which was selected when the import was initiated. E.g. if you right-click on a file in the **Navigation Area** and choose import, the imported files are placed immediately below the selected file. If you right-click a folder, the imported files are placed as the last file in that folder. If you right-click a folder, the imported files are placed as the last elements in this folder.

If you import one or more folders, the contents of the folder is automatically imported and placed in that folder in the **Navigation Area**. If the folder contains subfolders, the whole folder structure is imported.

In the import dialog (figure 7.1), there are three import options:

Automatic import This will import the file and *CLC Genomics Workbench* will try to determine the format of the file. The format is determined based on the file extension (e.g. SwissProt files have .swp at the end of the file name) in combination with a detection of elements in the file that are specific to the individual file formats. If the file type is not recognized, it will be imported as an external file. In most cases, automatic import will yield a successful result, but if the import goes wrong, the next option can be helpful:

Force import as type This option should be used if *CLC Genomics Workbench* cannot successfully determine the file format. By forcing the import as a specific type, the automatic determination of the file format is bypassed, and the file is imported as the type specified.

Force import as external file This option should be used if a file is imported as a bioinformatics file when it should just have been external file. It could be an ordinary text file which is imported as a sequence.

Import using drag and drop

It is also possible to drag a file from e.g. the desktop into the **Navigation Area** of *CLC Genomics Workbench*. This is equivalent to importing the file using the **Automatic import** option described above. If the file type is not recognized, it will be imported as an external file.

Import using copy/paste of text

If you have e.g. a text file or a browser displaying a sequence in one of the formats that can be imported by *CLC Genomics Workbench*, there is a very easy way to get this sequence into the **Navigation Area**:

> **Copy the text from the text file or browser | Select a folder in the Navigation Area | Paste ( )**

This will create a new sequence based on the text copied. This operation is equivalent to saving the text in a text file and importing it into the *CLC Genomics Workbench*.

If the sequence is not formatted, i.e. if you just have a text like this: "ATGACGAATAGGAGTTCTAGCTA" you can also paste this into the **Navigation Area**.

Note! Make sure you copy all the relevant text - otherwise *CLC Genomics Workbench* might not be able to interpret the text.

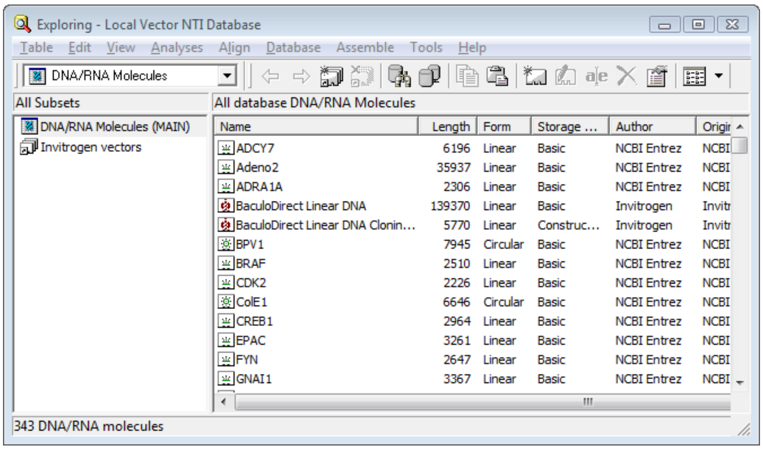
7.1.2 Import Vector NTI data

There are several ways of importing your Vector NTI data into the CLC Workbench. The best way to go depends on how your data is currently stored in Vector NTI:

- Your data is stored in the Vector NTI Local Database which can be accessed through Vector NTI Explorer. This is described in the first section below.
- Your data is stored as single files on your computer (just like Word documents etc.). This is described in the second section below.

Import from the Vector NTI Local Database

If your Vector NTI data are stored in a Vector NTI Local Database (as the one shown in figure 7.2), you can import all the data in one step, or you can import selected parts of it.


Exploring - Local Vector NTI Database
Table Edit View Analyses Align Database Assemble Tools Help
DNA/RNA Molecules
All Subsets
DNA/RNA Molecules (MAIN)
Invitrogen vectors
All database DNA/RNA Molecules

Name	Length	Form	Storage ...	Author	Origir
ADCY7	6196	Linear	Basic	NCBI Entrez	NCBI
Adeno2	35937	Linear	Basic	NCBI Entrez	NCBI
ADRA1A	2306	Linear	Basic	NCBI Entrez	NCBI
BaculoDirect Linear DNA	139370	Linear	Basic	Invitrogen	Invitr
BaculoDirect Linear DNA Clonin...	5770	Linear	Construc...	Invitrogen	Invitr
BPV1	7945	Circular	Basic	NCBI Entrez	NCBI
BRAF	2510	Linear	Basic	NCBI Entrez	NCBI
CDK2	2226	Linear	Basic	NCBI Entrez	NCBI
ColE1	6646	Circular	Basic	NCBI Entrez	NCBI
CREB1	2964	Linear	Basic	NCBI Entrez	NCBI
EPAC	3261	Linear	Basic	NCBI Entrez	NCBI
FYN	2647	Linear	Basic	NCBI Entrez	NCBI
GNAI1	3367	Linear	Basic	NCBI Entrez	NCBI

343 DNA/RNA molecules

Figure 7.2: *Data stored in the Vector NTI Local Database accessed through Vector NTI Explorer.*

Importing the entire database in one step

From the Workbench, there is a direct import of the whole database (see figure 7.3):

File | Import Vector NTI Database

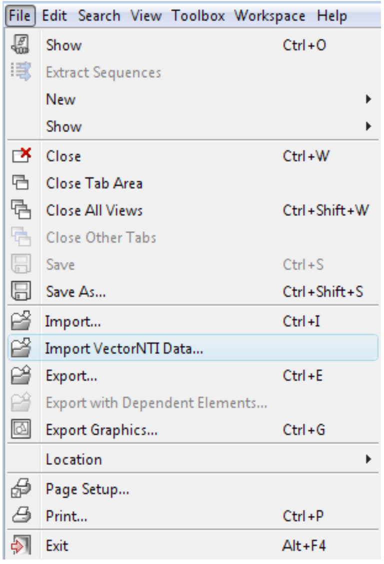


Figure 7.3: *Import the whole Vector NTI Database.*

This will bring up a dialog letting you choose to import from the default location of the database, or you can specify another location. If the database is installed in the default folder, like e.g. *C:\VNTI Database*, press **Yes**. If not, click **No** and specify the database folder manually.

When the import has finished, the data will be listed in the **Navigation Area** of the Workbench as shown in figure 7.4.

If something goes wrong during the import process, please report the problem to sup-

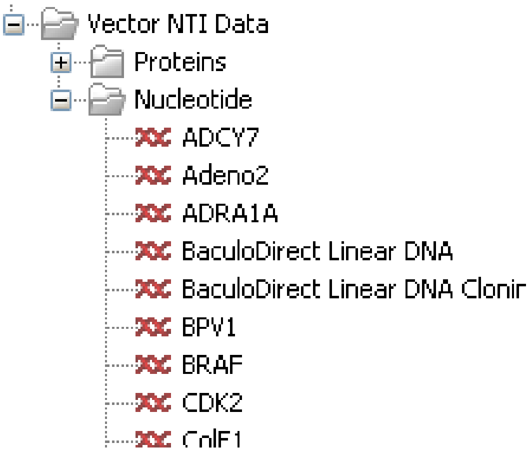
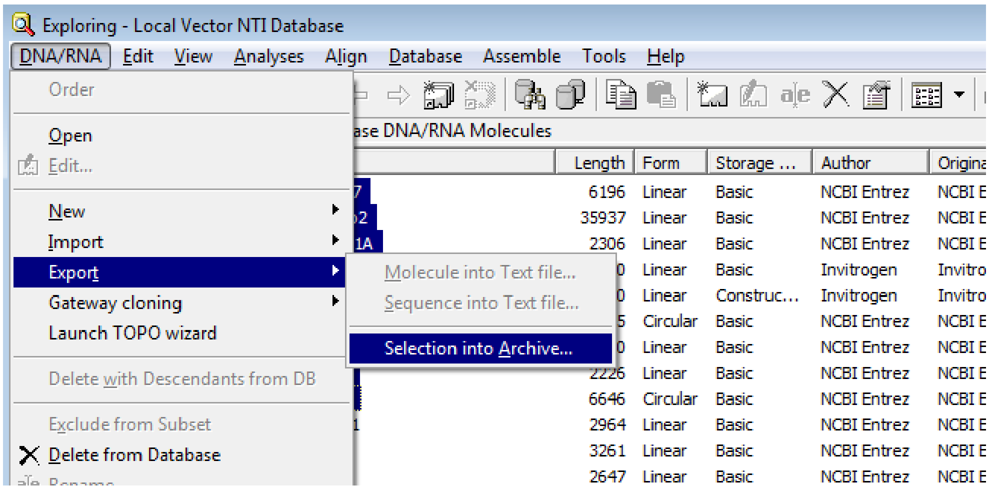


Figure 7.4: *The Vector NTI Data folder containing all imported sequences of the Vector NTI Database.*

port@clcbio.com. To circumvent the problem, see the following section on how to import parts of the database. It will take a few more steps, but you will most likely be able to import this way.

Importing parts of the database

Instead of importing the whole database automatically, you can export parts of the database from Vector NTI Explorer and subsequently import into the Workbench. First, export a selection of files as an archive as shown in figure 7.5.



	Length	Form	Storage ...	Author	Origina
7	6196	Linear	Basic	NCBI Entrez	NCBI E
2	35937	Linear	Basic	NCBI Entrez	NCBI E
1A	2306	Linear	Basic	NCBI Entrez	NCBI E
	0	Linear	Basic	Invitrogen	Invitro
	0	Linear	Construc...	Invitrogen	Invitro
	5	Circular	Basic	NCBI Entrez	NCBI E
	0	Linear	Basic	NCBI Entrez	NCBI E
	2226	Linear	Basic	NCBI Entrez	NCBI E
	6646	Circular	Basic	NCBI Entrez	NCBI E
1	2964	Linear	Basic	NCBI Entrez	NCBI E
	3261	Linear	Basic	NCBI Entrez	NCBI E
	2647	Linear	Basic	NCBI Entrez	NCBI E

Figure 7.5: *Select the relevant files and export them as an archive through the File menu.*

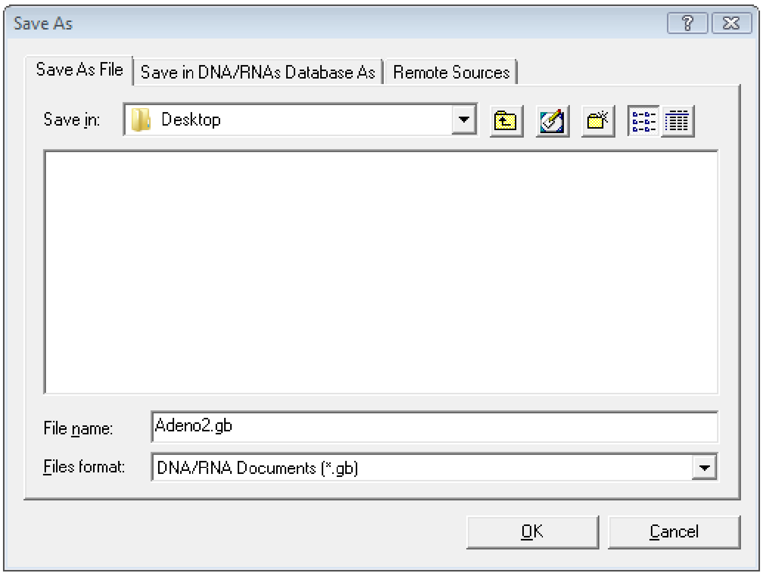
This will produce a file with a ma4-, pa4- or oa4-extension. Back in the CLC Workbench, click **Import** (📂) and select the file.

Importing single files

In Vector NTI, you can save a sequence in a file instead of in the database (see figure 7.6).

This will give you file with a .gb extension. This file can be easily imported into the CLC Workbench:

Import (📂) | **select the file** | **Select**



File name: Adeno2.gb

Files format: DNA/RNA Documents (*.gb)

Figure 7.6: *Saving a sequence as a file in Vector NTI.*

You don't have to import one file at a time. You can simply select a bunch of files or an entire folder, and the CLC Workbench will take care of the rest. Even if the files are in different formats.

You can also simply drag and drop the files into the **Navigation Area** of the CLC Workbench.

The Vector NTI import is a plug-in which is pre-installed in the Workbench. It can be uninstalled and updated using the plug-in manager (see section 1.7).

7.1.3 Export of bioinformatics data

CLC Genomics Workbench can export bioinformatic data in most of the formats that can be imported. There are a few exceptions. See section 7.1.1.

To export a file:

> **select the element to export | Export ( ) | choose where to export to | select 'File of type' | enter name of file | Save**

When exporting to CSV and tab delimited files, decimal numbers are formatted according to the Locale setting of the Workbench (see section 5.1). If you open the CSV or tab delimited file with spreadsheet software like Excel, you should make sure that both the Workbench and the spreadsheet software are using the same Locale.

Note! The **Export** dialog decides which types of files you are allowed to export into, depending on what type of data you want to export. E.g. protein sequences can be exported into GenBank, Fasta, Swiss-Prot and CLC-formats.

Export of folders and multiple elements

The .zip file type can be used to export all kinds of files and is therefore especially useful in these situations:

- Export of one or more folders including all underlying elements and folders.

- If you want to export two or more elements into one file.

Export of folders is similar to export of single files. Exporting multiple files (of different formats) is done in .zip-format. This is how you export a folder:

select the folder to export | Export () | choose where to export to | enter name | Save

You can export multiple files of the same type into formats other than ZIP (.zip). E.g. two DNA sequences can be exported in GenBank format:

select the two sequences by <Ctrl>-click (⌘ -click on Mac) or <Shift>-click | Export () | choose where to export to | choose GenBank (.gbk) format | enter name the new file | Save

Export of dependent elements

When exporting e.g. an alignment, *CLC Genomics Workbench* can export the alignment including all the sequences that were used to create it. This way, when sending your alignment (with the dependent sequences), your colleagues can reproduce your findings with adjusted parameters, if desired. To export with dependent files:

select the element in Navigation Area | File in Menu Bar | Export with Dependent Elements | enter name of of the new file | choose where to export to | Save

The result is a folder containing the exported file with dependent elements, stored automatically in a folder on the desired location of your desk.

Export history

To export an element's history:

select the element in Navigation Area Export () | select History PDF(.pdf) | choose where to export to | Save

The entire history of the element is then exported in pdf format.

The CLC format

CLC Genomics Workbench keeps all bioinformatic data in the CLC format. Compared to other formats, the CLC format contains more information about the object, like its history and comments. The CLC format is also able to hold several elements of different types (e.g. an alignment, a graph and a phylogenetic tree). This means that if you are exporting your data to another CLC Workbench, you can use the CLC format to export several elements in one file, and you will preserve all the information.

Note! CLC files can be exported from and imported into all the different CLC Workbenches.

Backup

If you wish to secure your data from computer breakdowns, it is advisable to perform regular backups of your data. Backing up data in the *CLC Genomics Workbench* is done in two ways:

- Making a backup of each of the folders represented by the locations in the **Navigation Area**.
- Selecting all locations in the **Navigation Area** and export () in .zip format. The resulting file will contain all the data stored in the **Navigation Area** and can be imported into *CLC Genomics Workbench* if you wish to restore from the back-up at some point.

No matter which method is used for backup, you may have to re-define the locations in the **Navigation Area** if you restore your data from a computer breakdown.

7.2 External files

In order to help you organize your research projects, *CLC Genomics Workbench* lets you import all kinds of files. E.g. if you have Word, Excel or pdf-files related to your project, you can import them into the **Navigation Area** of *CLC Genomics Workbench*. Importing an external file creates a copy of the file which is stored at the location you have chosen for import. The file can now be opened by double-clicking the file in the **Navigation Area**. The file is opened using the default application for this file type (e.g. Microsoft Word for .doc-files and Adobe Reader for .pdf).

External files are imported and exported in the same way as bioinformatics files (see section 7.1.1). Bioinformatics files not recognized by *CLC Genomics Workbench* are also treated as external files.

7.3 Export graphics to files

CLC Genomics Workbench supports export of graphics into a number of formats. This way, the visible output of your work can easily be saved and used in presentations, reports etc. The **Export Graphics** function () is found in the **Toolbar**.

CLC Genomics Workbench uses a WYSIWYG principle for graphics export: What You See Is What You Get. This means that you should use the options in the Side Panel to change how your data, e.g. a sequence, looks in the program. When you export it, the graphics file will look exactly the same way.

It is not possible to export graphics of elements directly from the **Navigation Area**. They must first be opened in a view in order to be exported. To export graphics of the contents of a view:

select tab of View | Graphics () on Toolbar

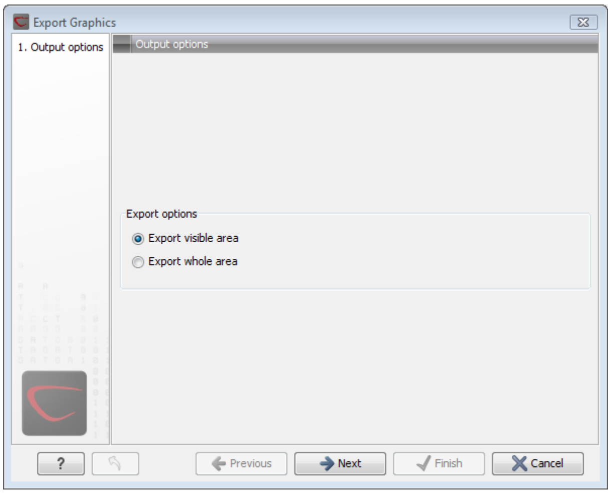
This will display the dialog shown in figure 7.7.

7.3.1 Which part of the view to export

In this dialog you can choose to:

- **Export visible area**, or
- **Export whole view**

These options are available for all views that can be zoomed in and out. In figure 7.8 is a view of a circular sequence which is zoomed in so that you can only see a part of it.



Export options

● Export visible area

○ Export whole area

Figure 7.7: *Selecting to export whole view or to export only the visible area.*

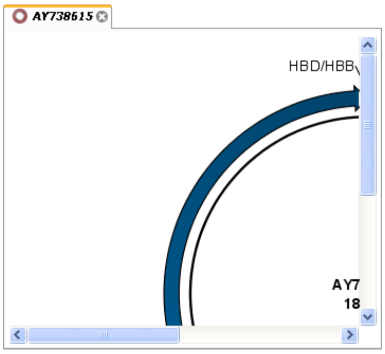


Figure 7.8: *A circular sequence as it looks on the screen.*

When selecting **Export visible area**, the exported file will only contain the part of the sequence that is *visible* in the view. The result from exporting the view from figure 7.8 and choosing **Export visible area** can be seen in figure 7.9.

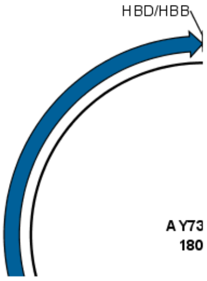


Figure 7.9: *The exported graphics file when selecting Export visible area.*

On the other hand, if you select **Export whole view**, you will get a result that looks like figure 7.10. This means that the graphics file will also include the part of the sequence which is not visible when you have zoomed in.

For 3D structures, this first step is omitted and you will always export what is shown in the view (equivalent to selecting **Export visible area**).

Click **Next** when you have chosen which part of the view to export.

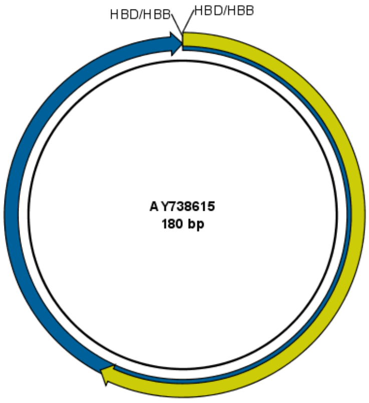
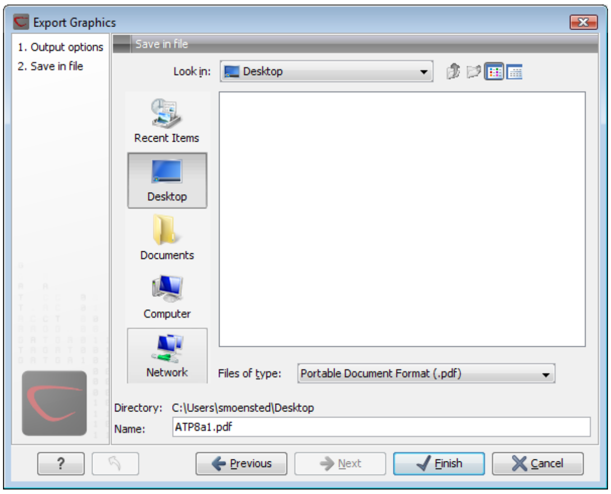


Figure 7.10: *The exported graphics file when selecting Export whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*

7.3.2 Save location and file formats

In this step, you can choose name and save location for the graphics file (see figure 7.11).



Look in: Desktop

Files of type: Portable Document Format (.pdf)

Directory: C:\Users\smoensted\Desktop

Name: ATP8a1.pdf

Figure 7.11: *Location and name for the graphics file.*

CLC Genomics Workbench supports the following file formats for graphics export:

Format	**Suffix**	**Type**
Portable Network Graphics	.png	bitmap
JPEG	.jpg	bitmap
Tagged Image File	.tif	bitmap
PostScript	.ps	vector graphics
Encapsulated PostScript	.eps	vector graphics
Portable Document Format	.pdf	vector graphics
Scalable Vector Graphics	.svg	vector graphics

These formats can be divided into bitmap and vector graphics. The difference between these two categories is described below:

Bitmap images

In a bitmap image, each dot in the image has a specified color. This implies, that if you zoom in on the image there will not be enough dots, and if you zoom out there will be too many. In these cases the image viewer has to interpolate the colors to fit what is actually looked at. A bitmap image needs to have a high resolution if you want to zoom in. This format is a good choice for storing images without large shapes (e.g. dot plots). It is also appropriate if you don't have the need for resizing and editing the image after export.

Vector graphics

Vector graphic is a collection of shapes. Thus what is stored is e.g. information about where a line starts and ends, and the color of the line and its width. This enables a given viewer to decide how to draw the line, no matter what the zoom factor is, thereby always giving a correct image. This format is good for e.g. graphs and reports, but less usable for e.g. dot plots. If the image is to be resized or edited, vector graphics are by far the best format to store graphics. If you open a vector graphics file in an application like e.g. Adobe Illustrator, you will be able to manipulate the image in great detail.

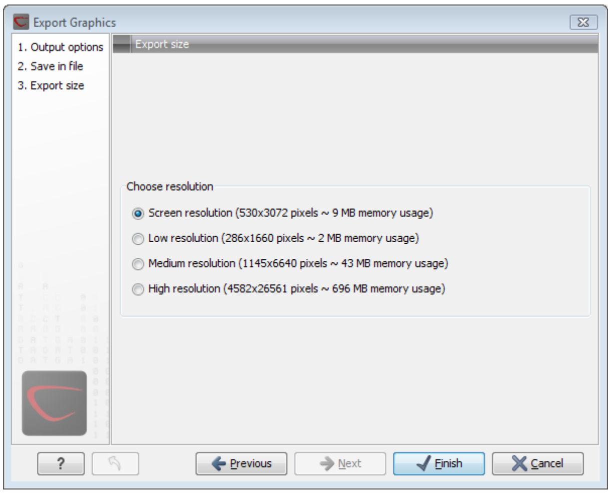
Graphics files can also be imported into the **Navigation Area**. However, no kinds of graphics files can be displayed in *CLC Genomics Workbench*. See section 7.2 for more about importing external files into *CLC Genomics Workbench*.

7.3.3 Graphics export parameters

When you have specified the name and location to save the graphics file, you can either click **Next** or **Finish**. Clicking **Next** allows you to set further parameters for the graphics export, whereas clicking **Finish** will export using the parameters that you have set last time you made a graphics export in that file format (if it is the first time, it will use default parameters).

Parameters for bitmap formats

For bitmap files, clicking **Next** will display the dialog shown in figure 7.12.



Choose resolution

- ● Screen resolution (530x3072 pixels ~ 9 MB memory usage)
- ○ Low resolution (286x1660 pixels ~ 2 MB memory usage)
- ○ Medium resolution (1145x6640 pixels ~ 43 MB memory usage)
- ○ High resolution (4582x26561 pixels ~ 696 MB memory usage)

Figure 7.12: *Parameters for bitmap formats: size of the graphics file.*

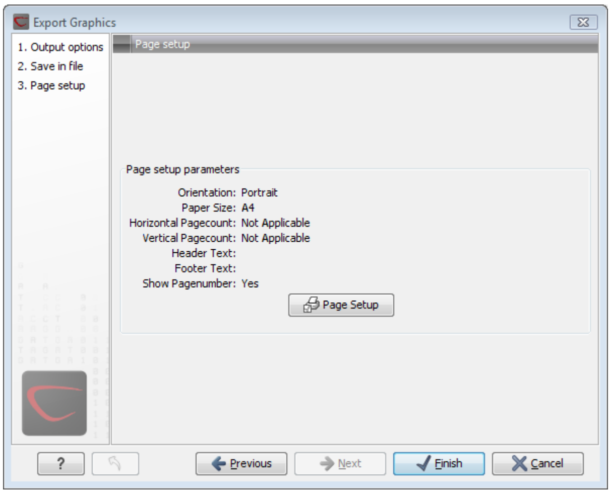
You can adjust the size (the resolution) of the file to four standard sizes:

- Screen resolution
- Low resolution
- Medium resolution
- High resolution

The actual size in pixels is displayed in parentheses. An estimate of the memory usage for exporting the file is also shown. If the image is to be used on computer screens only, a low resolution is sufficient. If the image is going to be used on printed material, a higher resolution is necessary to produce a good result.

Parameters for vector formats

For pdf format, clicking **Next** will display the dialog shown in figure 7.13 (this is only the case if the graphics is using more than one page).



Page setup parameters

Orientation: Portrait
Paper Size: A4
Horizontal Pagecount: Not Applicable
Vertical Pagecount: Not Applicable
Header Text:
Footer Text:
Show Pagenumber: Yes

Figure 7.13: *Page setup parameters for vector formats.*

The settings for the page setup are shown, and clicking the **Page Setup** button will display a dialog where these settings can ba adjusted. This dialog is described in section 6.2.

The page setup is only available if you have selected to export the whole view - if you have chosen to export the visible area only, the graphics file will be on one page with no headers or footers.

7.3.4 Exporting protein reports

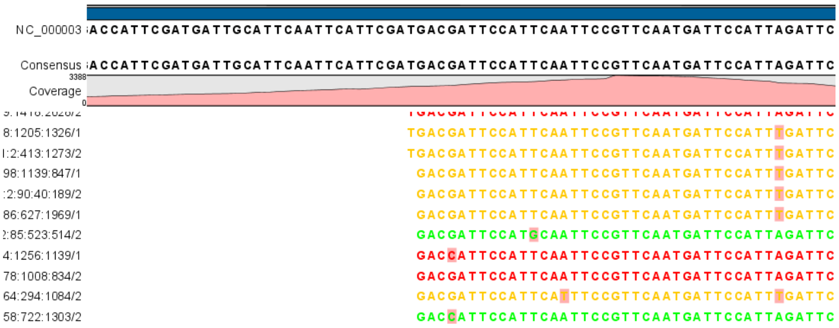
It is possible to export a protein report using the normal **Export** function (📂) which will generate a pdf file with a table of contents:

Click the report in the Navigation Area | Export (📂) in the Toolbar | select pdf

You can also choose to export a protein report using the **Export graphics** function (🖼), but in this way you will not get the table of contents.

7.4 Export graph data points to a file

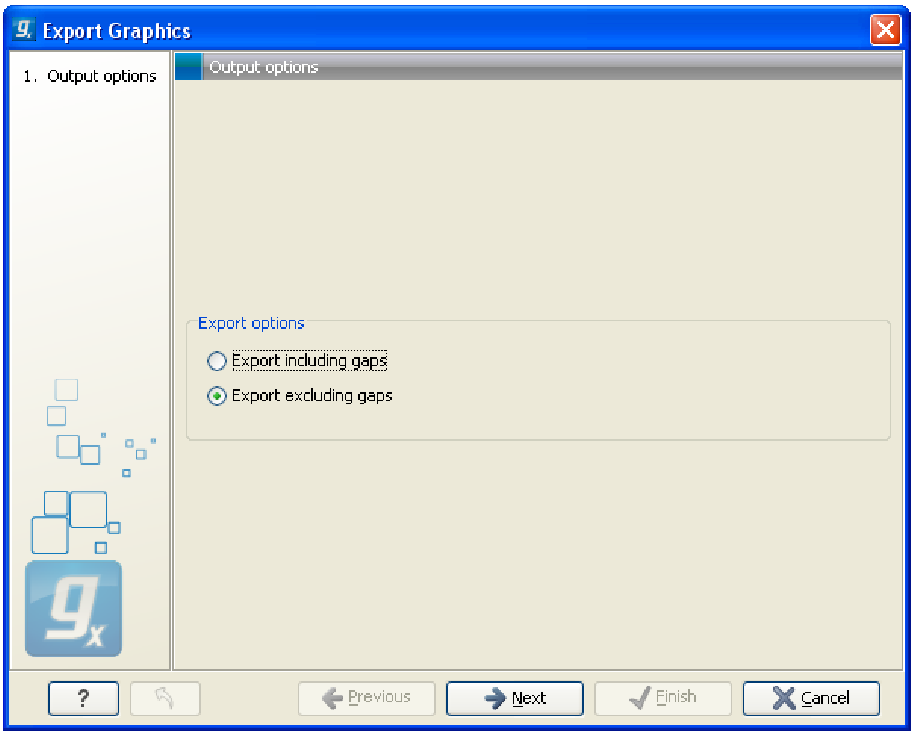
Data points for graphs displayed along the sequence or along an alignment, mapping or BLAST result, can be exported to a semicolon-separated text file (csv format). An example of such a graph is shown in figure 7.14. This graph shows the coverage of reads of a read mapping (produced with *CLC Genomics Workbench*).



```
NC_000003        ACCATTCGATGATTGCATTCAATTCATTCGATGACGATTCCATTCAATTCCGTTCAATGATTCCATTAGATTC

Consensus        ACCATTCGATGATTGCATTCAATTCATTCGATGACGATTCCATTCAATTCCGTTCAATGATTCCATTAGATTC
            3388
Coverage
               0
8:1205:1326/1                                   TGACGATTCCATTCAATTCCGTTCAATGATTCCATTTGATTC
I:2:413:1273/2                                  TGACGATTCCATTCAATTCCGTTCAATGATTCCATTTGATTC
98:1139:847/1                                    GACGATTCCATTCAATTCCGTTCAATGATTCCATTTGATTC
:2:90:40:189/2                                   GACGATTCCATTCAATTCCGTTCAATGATTCCATTTGATTC
86:627:1969/1                                    GACGATTCCATTCAATTCCGTTCAATGATTCCATTTGATTC
2:85:523:514/2                                   GACGATTCCATGCAATTCCGTTCAATGATTCCATTAGATTC
4:1256:1139/1                                    GACCATTCCATTCAATTCCGTTCAATGATTCCATTAGATTC
78:1008:834/2                                    GACGATTCCATTCAATTCCGTTCAATGATTCCATTAGATTC
64:294:1084/2                                    GACGATTCCATTCATTTCCGTTCAATGATTCCATTTGATTC
58:722:1303/2                                    GACCATTCCATTCAATTCCGTTCAATGATTCCATTAGATTC
```

Figure 7.14: *A graph displayed along the mapped reads. Right-click the graph to export the data points to a file.*

To export the data points for the graph, right-click the graph and choose **Export Graph to Comma-separated File**. Depending on what kind of graph you have selected, different options will be shown: If the graph is covering a set of aligned sequences with a main sequence, such as read mappings and BLAST results, the dialog shown in figure 7.15 will be displayed. These kinds of graphs are located under **Alignment info** in the Side Panel. In all other cases, a normal file dialog will be shown letting you specify name and location for the file.



Export Graphics

1. Output options

Output options

Export options

○ Export including gaps

● Export excluding gaps

Figure 7.15: *Choosing to include data points with gaps*

In this dialog, select whether you wish to include positions where the main sequence (the reference sequence for read mappings and the query sequence for BLAST results) has gaps. If you are exporting e.g. coverage information from a read mapping, you would probably want to exclude gaps, if you want the positions in the exported file to match the reference (i.e. chromosome) coordinates. If you export including gaps, the data points in the file no longer corresponds to the reference coordinates, because each gap will shift the coordinates.

Clicking **Next** will present a file dialog letting you specify name and location for the file.

The output format of the file is like this:

```
"Position";"Value";
"1";"13";
"2";"16";
"3";"23";
"4";"17";
...
```

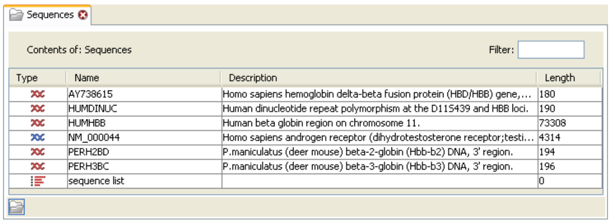
7.5 Copy/paste view output

The content of tables, e.g. in reports, folder lists, and sequence lists can be copy/pasted into different programs, where it can be edited. *CLC Genomics Workbench* pastes the data in tabulator separated format which is useful if you use programs like Microsoft Word and Excel. There is a huge number of programs in which the copy/paste can be applied. For simplicity, we include one example of the copy/paste function from a **Folder Content** view to Microsoft Excel.

First step is to select the desired elements in the view:

> **click a line in the Folder Content view | hold Shift-button | press arrow down/up key**

See figure 7.16.



Contents of: Sequences

Filter:

Type	Name	Description	Length
	AY738615	Homo sapiens hemoglobin delta-beta fusion protein (HBD/HBB) gene,...	180
	HUMDINUC	Human dinucleotide repeat polymorphism at the D11S439 and HBB loci.	190
	HUMHBB	Human beta globin region on chromosome 11.	73308
	NM_000044	Homo sapiens androgen receptor (dihydrotestosterone receptor;testi...	4314
	PERH2BD	P.maniculatus (deer mouse) beta-2-globin (Hbb-b2) DNA, 3' region.	194
	PERH3BC	P.maniculatus (deer mouse) beta-3-globin (Hbb-b3) DNA, 3' region.	196
	sequence list		0

Figure 7.16: *Selected elements in a Folder Content view.*

When the elements are selected, do the following to copy the selected elements:

> **right-click one of the selected elements | Edit | Copy ( )**

Then:

> **right-click in the cell A1 | Paste ( )**

The outcome might appear unorganized, but with a few operations the structure of the view in *CLC Genomics Workbench* can be produced. (Except the icons which are replaced by file references in Excel.)

Note that all tables can also be **Exported** ( ) directly in Excel format.

Chapter 8

History log

Contents

CLC Genomics Workbench keeps a log of all operations you make in the program. If e.g. you rename a sequence, align sequences, create a phylogenetic tree or translate a sequence, you can always go back and check what you have done. In this way, you are able to document and reproduce previous operations.

This can be useful in several situations: It can be used for documentation purposes, where you can specify exactly how your data has been created and modified. It can also be useful if you return to a project after some time and want to refresh your memory on how the data was created. Also, if you have performed an analysis and you want to reproduce the analysis on another element, you can check the history of the analysis which will give you all parameters you set.

This chapter will describe how to use the **History** functionality of *CLC Genomics Workbench*.

8.1 Element history

You can view the history of all elements in the **Navigation Area** except files that are opened in other programs (e.g. Word and pdf-files). The history starts when the element appears for the first time in *CLC Genomics Workbench*. To view the history of an element:

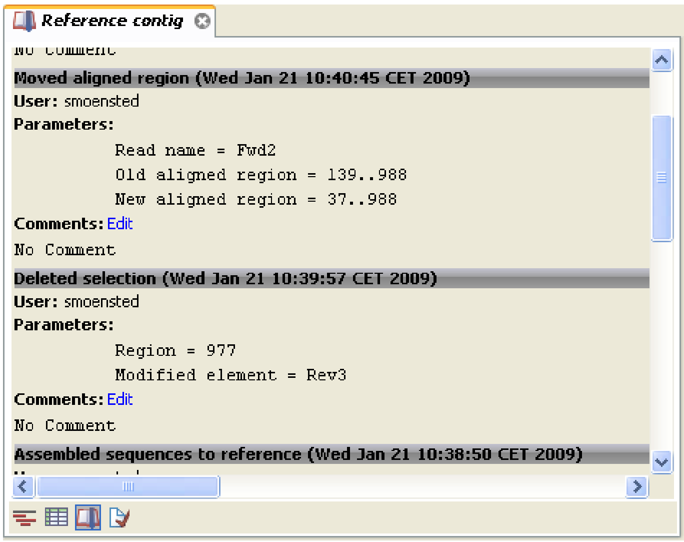
Select the element in the Navigation Area | Show ( ) in the Toolbar | History ( )

or **If the element is already open | History ( ) at the bottom left part of the view**

This opens a view that looks like the one in figure 8.1.

When opening an element's history is opened, the newest change is submitted in the top of the view. The following information is available:

- **Title**. The action that the user performed.
- **Date and time**. Date and time for the operation. The date and time are displayed according



Reference contig

```
No Comment
Moved aligned region (Wed Jan 21 10:40:45 CET 2009)
User: smoensted
Parameters:
          Read name = Fwd2
          Old aligned region = 139..988
          New aligned region = 37..988
Comments: Edit
No Comment
Deleted selection (Wed Jan 21 10:39:57 CET 2009)
User: smoensted
Parameters:
          Region = 977
          Modified element = Rev3
Comments: Edit
No Comment
Assembled sequences to reference (Wed Jan 21 10:38:50 CET 2009)
```

Figure 8.1: *An element's history.*

to your locale settings (see section 5.1).

- **User**. The user who performed the operation. If you import some data created by another person in a CLC Workbench, that persons name will be shown.
- **Parameters**. Details about the action performed. This could be the parameters that was chosen for an analysis.
- **Origins from**. This information is usually shown at the bottom of an element's history. Here, you can see which elements the current element origins from. If you have e.g. created an alignment of three sequences, the three sequences are shown here. Clicking the element selects it in the **Navigation Area**, and clicking the 'history' link opens the element's own history.
- **Comments**. By clicking **Edit** you can enter your own comments regarding this entry in the history. These comments are saved.

8.1.1 Sharing data with history

The history of an element is attached to that element, which means that exporting an element in CLC format (*.clc) will export the history too. In this way, you can share folders and files with others while preserving the history. If an element's history includes source elements (i.e. if there are elements listed in 'Origins from'), they must also be exported in order to see the full history. Otherwise, the history will have entries named "Element deleted". An easy way to export an element with all its source elements is to use the **Export Dependent Elements** function described in section 7.1.3.

The history view can be printed. To do so, click the **Print** icon (🖨). The history can also be exported as a pdf file:

Select the element in the Navigation Area | Export (📂) | in "File of type" choose History PDF | Save

Chapter 9

Batching and result handling

Contents

9.1 Batch processing

Most of the analyses in the **Toolbox** are able to perform the same analysis on several elements in one batch. This means that analyzing large amounts of data is very easily accomplished. As an example, if you use the **Find Binding Sites and Create Fragments ( )** tool, if you supply five sequences as shown in figure 9.1, the result table will present an overview of the results for all five sequences.

This is because the input sequences are pooled before running the analysis. If you want individual outputs for each sequence, you would need to run the tool five times, or alternatively use the **Batching mode**.

Batching mode is activated by clicking the **Batch** checkbox in dialog where the input data is selected. Batching simply means that each data set is run separately, just as if the tool has been run manually for each one. For some analyses, this simply means that each input sequence should be run separately, but in other cases it is desirable to pool sets of files together in one run. This selection of data for a batch run is defined as a **batch unit**.

When batching is selected, the data to be added is the folder containing the data you want to batch. The content of the folder is assigned into batch units based on this concept:

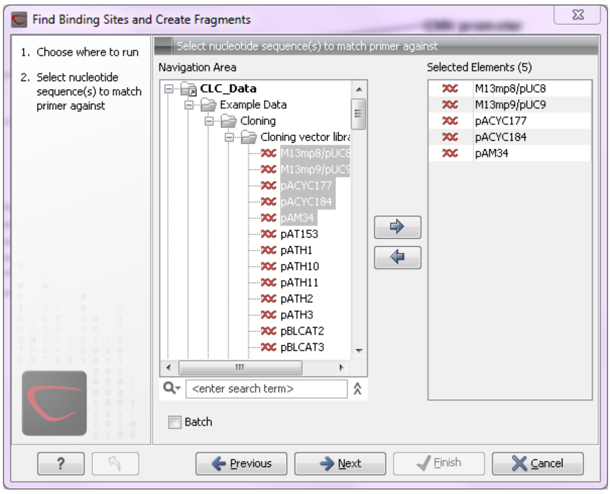


Figure 9.1: *Inputting five sequences to Find Binding Sites and Create Fragments.*

- All subfolders are treated as individual batch units. This means that if the subfolder contains several input files, they will be pooled as one batch unit. Nested subfolders (i.e. subfolders within the subfolder) are ignored.
- All files that are not in subfolders are treated as individual batch units.

An example of a batch run is shown in figure 9.2.

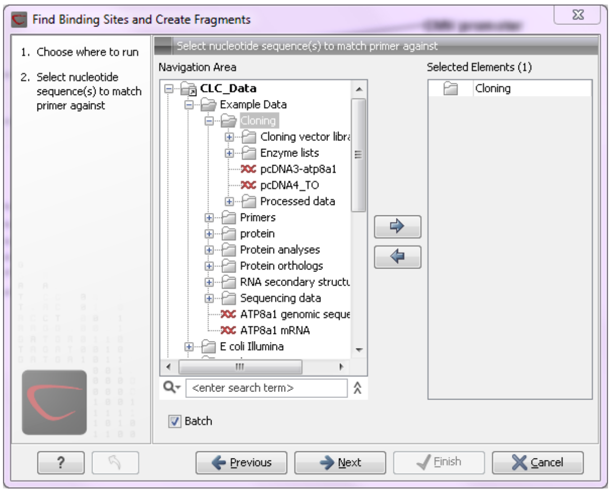


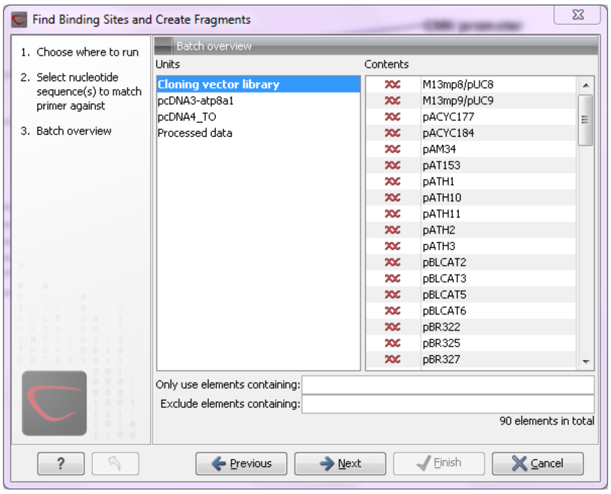
Figure 9.2: *The Cloning folder includes both folders and sequences.*

The `Cloning` folder that is found in the example data (see section 1.6.2) contains two sequences (✗✗) and three folders (🗀). If you click **Batch**, only folders can be added to the list of selected elements in the right-hand side of the dialog. To run the contents of the Cloning folder in batch, double-click to select it.

When the Cloning folder is selected and you click **Next**, a batch overview is shown.

9.1.1 Batch overview

The batch overview lists the batch units to the left and the contents of the selected unit to the right (see figure 9.3).



1. Choose where to run
2. Select nucleotide sequence(s) to match primer against
3. Batch overview

Units:
Cloning vector library
pcDNA3-atp8a1
pcDNA4_TO
Processed data

Contents:
M13mp8/pUC8
M13mp9/pUC9
pACYC177
pACYC184
pAM34
pAT153
pATH1
pATH10
pATH11
pATH2
pATH3
pBLCAT2
pBLCAT3
pBLCAT5
pBLCAT6
pBR322
pBR325
pBR327

Only use elements containing:
Exclude elements containing:
90 elements in total

Figure 9.3: *Overview of the batch run.*

In this example, the two sequences are defined as separate batch units because they are located at the top level of the Cloning folder. There were also three folders in the Cloning folder (see figure 9.2), and two of them are listed as well. This means that the contents of these folders are pooled in one batch run (you can see the contents of the `Cloning vector library` batch run in the panel at the right-hand side of the dialog). The reason why the `Enzyme lists` folder is not listed as a batch unit is that it does not contain any sequences.

In this overview dialog, the Workbench has filtered the data so that only the types of data accepted by the tool is shown (DNA sequences in the example above).

9.1.2 Batch filtering and counting

At the bottom of the dialog shown in figure 9.3, the Workbench counts the number of files that will be run in total (90 in this case). This is counted across all the batch units.

In some situations it is useful to filter the input for the batching based on names. As an example, this could be to include only paired reads for a mapping, by only allowing names where "paired" is part of the name.

This is achieved using the **Only use elements containing** and **Exclude elements containing** text fields. Note that the count is dynamically updated to reflect the number of input files based on the filtering.

If a complete batch unit should be removed, you can select it, right-click and choose **Remove Batch Unit**. You can also remove items from the contents of each batch unit using right-click and **Remove Element**.

9.1.3 Setting parameters for batch runs

For some tools, the subsequent dialogs depend on the input data. In this case, one of the units is specified as parameter prototype and will be used to guide the choices in the dialogs. Per default, this will be the first batch unit (marked in bold), but this can be changed by right-clicking another batch unit and click **Set as Parameter Prototype**.

Note that the Workbench is validating a lot of the input and parameters when running in normal "non-batch" mode. When running in batch, this validation is not performed, and this means that some analyses will fail if combinations of input data and parameters are not right. Therefore batching should only be used when the batch units are very homogenous in terms of the type and size of data.

9.1.4 Running the analysis and organizing the results

At the last dialog before clicking **Finish**, it is only possible to use the **Save** option. When a tool is run in batch mode, it will place the result files in the same folder as the input files. In the example shown in figure 9.3, the result of the two single sequences will be placed in the Cloning folder, whereas the results for the `Cloning vector library` and `Processed data` runs will be placed inside these folders.

When the batch run is started, there will be one "master" process representing the overall batch job, and there will then be a separate process for each batch unit. The behavior of this is different between Workbench and Server:

- When running the batch job in the Workbench, only one batch unit is run at a time. So when the first batch unit is done, the second will be started and so on. This is done in order to avoid many parallel analyses that would draw on the same compute resources and slow down the computer.
- When this is run on a CLC Server (see `http://clcbio.com/server`), all the processes are placed in the queue, and the queue is then taking care of distributing the jobs. This means that if the server set-up includes multiple nodes, the jobs can be run in parallel.

If you need to stop the whole batch run, you need to stop the "master" process.

9.1.5 Running de novo assembly and read mapping in batch

De novo assembly and read mapping are special in batch mode because they usually have the option of assigning individual mapping parameters to each input file. When running in batch mode this is not possible. Instead, you can change the default parameters used for long and short reads, respectively. You can also set the paired distance for paired data.

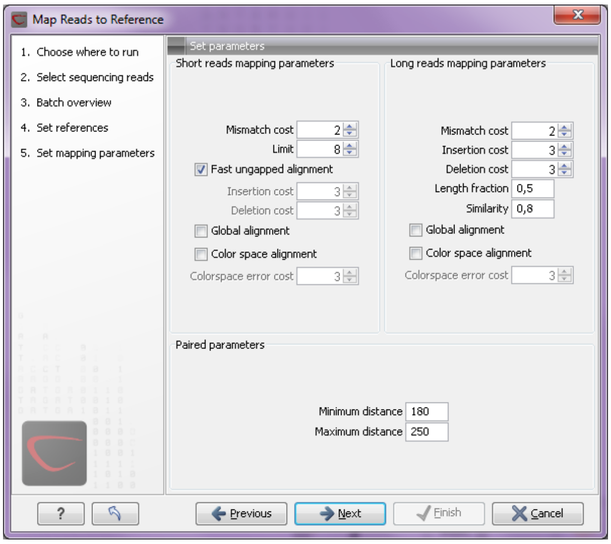
Note that this means that you cannot use a combination of paired-end and mate-pair data for batching.

Figure 9.4 shows the parameter dialog when running read mapping in batch.

Note that you can only specify one setting for all short reads, and one setting for all long reads. When the analysis is run, the reads are automatically categorized as either long or short, and the parameters specified in the dialog are applied. The same goes for all reads that are imported as paired where the minimum and maximum distances are applied.

9.2 How to handle results of analyses

This section will explain how results generated from tools in the Toolbox are handled by *CLC Genomics Workbench*. Note that this also applies to tools not running in batch mode (see above). All the analyses in the **Toolbox** are performed in a step-by-step procedure. First, you select



Map Reads to Reference

1. Choose where to run
2. Select sequencing reads
3. Batch overview
4. Set references
5. Set mapping parameters

Set parameters

Short reads mapping parameters

Mismatch cost 2
Limit 8
☑ Fast ungapped alignment
Insertion cost 3
Deletion cost 3
□ Global alignment
□ Color space alignment
Colorspace error cost 3

Long reads mapping parameters

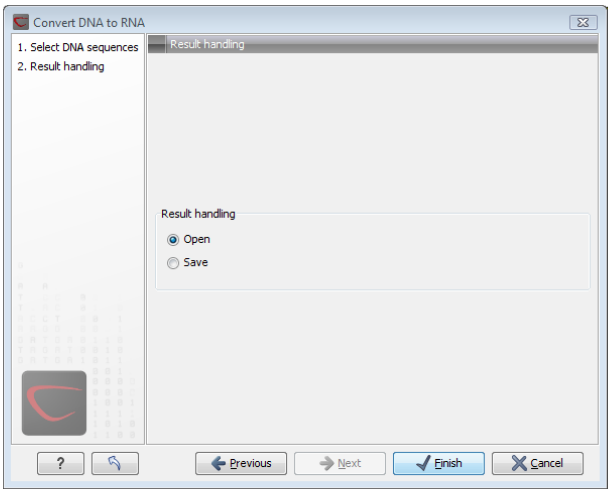
Mismatch cost 2
Insertion cost 3
Deletion cost 3
Length fraction 0,5
Similarity 0,8
□ Global alignment
□ Color space alignment
Colorspace error cost 3

Paired parameters

Minimum distance 180
Maximum distance 250

Figure 9.4: *Read mapping parameters in batch.*

elements for analyses, and then there are a number of steps where you can specify parameters (some of the analyses have no parameters, e.g. when translating DNA to RNA). The final step concerns the handling of the results of the analysis, and it is almost identical for all the analyses so we explain it in this section in general.



Convert DNA to RNA

1. Select DNA sequences
2. Result handling

Result handling

Result handling
● Open
○ Save

Figure 9.5: *The last step of the analyses exemplified by Translate DNA to RNA.*

In this step, shown in figure 9.5, you have two options:

- **Open.** This will open the result of the analysis in a view. This is the default setting.
- **Save.** This means that the result will not be opened but saved to a folder in the **Navigation Area**. If you select this option, click **Next** and you will see one more step where you can specify where to save the results (see figure 9.6). In this step, you also have the option of creating a new folder or adding a location by clicking the buttons ( )/ ( ) at the top of the dialog.

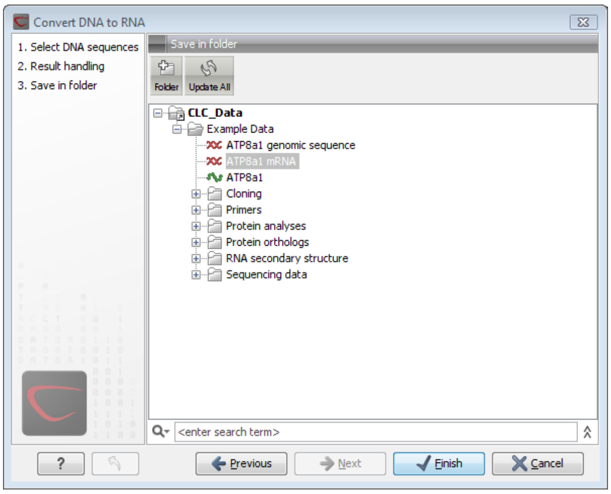
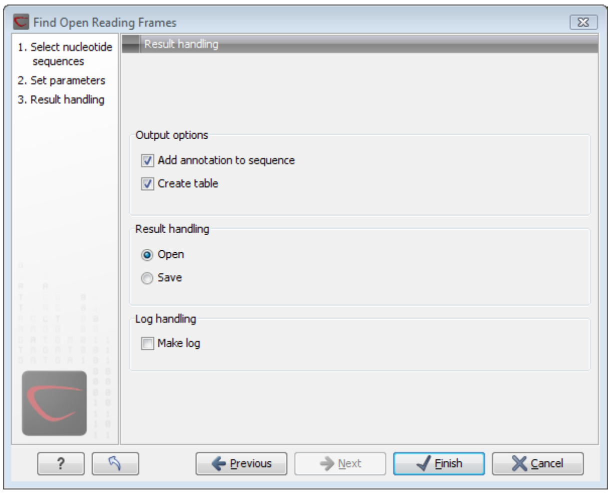


Figure 9.6: *Specify a folder for the results of the analysis.*

9.2.1 Table outputs

Some analyses also generate a table with results, and for these analyses the last step looks like figure 9.7.



Output options
☑ Add annotation to sequence
☑ Create table

Result handling
● Open
○ Save

Log handling
□ Make log

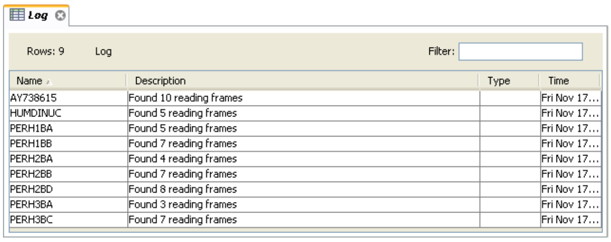
Figure 9.7: *Analyses which also generate tables.*

In addition to the **Open** and **Save** options you can also choose whether the result of the analysis should be added as annotations on the sequence or shown on a table. If both options are selected, you will be able to click the results in the table and the corresponding region on the sequence will be selected.

If you choose to add annotations to the sequence, they can be removed afterwards by clicking **Undo** (↶) in the **Toolbar**.

9.2.2 Batch log

For some analyses, there is an extra option in the final step to create a log of the batch process (see e.g. figure 9.7). This log will be created in the beginning of the process and continually updated with information about the results. See an example of a log in figure 9.8. In this example, the log displays information about how many open reading frames were found.



Rows: 9 Log Filter:

Name	Description	Type	Time
AY738615	Found 10 reading frames		Fri Nov 17...
HUMDINUC	Found 5 reading frames		Fri Nov 17...
PERH1BA	Found 5 reading frames		Fri Nov 17...
PERH1BB	Found 7 reading frames		Fri Nov 17...
PERH2BA	Found 4 reading frames		Fri Nov 17...
PERH2BB	Found 7 reading frames		Fri Nov 17...
PERH2BD	Found 8 reading frames		Fri Nov 17...
PERH3BA	Found 3 reading frames		Fri Nov 17...
PERH3BC	Found 7 reading frames		Fri Nov 17...

Figure 9.8: *An example of a batch log when finding open reading frames.*

The log will either be saved with the results of the analysis or opened in a view with the results, depending on how you chose to handle the results.

Part III

Bioinformatics

Chapter 10

Viewing and editing sequences

Contents

CLC Genomics Workbench offers five different ways of viewing and editing single sequences as described in the first five sections of this chapter. Furthermore, this chapter also explains how to create a new sequence and how to gather several sequences in a sequence list.

10.1 View sequence

When you double-click a sequence in the **Navigation Area**, the sequence will open automatically, and you will see the nucleotides or amino acids. The zoom options described in section 3.3 allow you to e.g. zoom out in order to see more of the sequence in one view. There are a number of options for viewing and editing the sequence which are all described in this section. All the options described in this section also apply to alignments (further described in section 22.2).

10.1.1 Sequence settings in Side Panel

Each view of a sequence has a **Side Panel** located at the right side of the view (see figure 10.1.

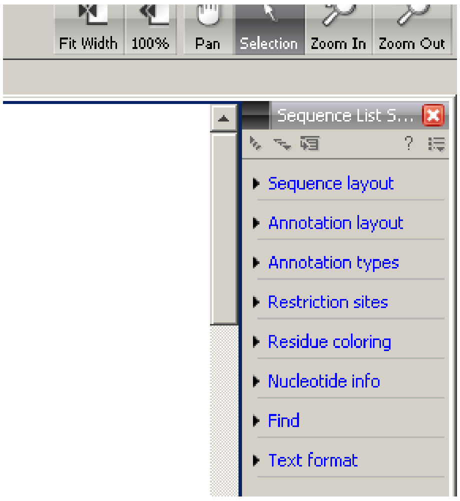


Figure 10.1: *Overview of the Side Panel which is always shown to the right of a view.*

When you make changes in the **Side Panel** the view of the sequence is instantly updated. To show or hide the **Side Panel**:

select the View | Ctrl + U

or **Click the (☒) at the top right corner of the Side Panel to hide | Click the gray Side Panel button to the right to show**

Below, each group of settings will be explained. Some of the preferences are not the same for nucleotide and protein sequences, but the differences will be explained for each group of settings.

Note! When you make changes to the settings in the **Side Panel**, they are not automatically saved when you save the sequence. Click **Save/restore Settings** (☰) to save the settings (see section 5.6 for more information).

Sequence Layout

These preferences determine the overall layout of the sequence:

- **Spacing.** Inserts a space at a specified interval:
  - **No spacing.** The sequence is shown with no spaces.
  - **Every 10 residues.** There is a space every 10 residues, starting from the beginning of the sequence.
  - **Every 3 residues, frame 1.** There is a space every 3 residues, corresponding to the reading frame starting at the first residue.

  - **Every 3 residues, frame 2.** There is a space every 3 residues, corresponding to the reading frame starting at the second residue.
  - **Every 3 residues, frame 3.** There is a space every 3 residues, corresponding to the reading frame starting at the third residue.

- **Wrap sequences.** Shows the sequence on more than one line.
  - **No wrap.** The sequence is displayed on one line.
  - **Auto wrap.** Wraps the sequence to fit the width of the view, not matter if it is zoomed in our out (displays minimum 10 nucleotides on each line).
  - **Fixed wrap.** Makes it possible to specify when the sequence should be wrapped. In the text field below, you can choose the number of residues to display on each line.

- **Double stranded.** Shows both strands of a sequence (only applies to DNA sequences).

- **Numbers on sequences.** Shows residue positions along the sequence. The starting point can be changed by setting the number in the field below. If you set it to e.g. 101, the first residue will have the position of -100. This can also be done by right-clicking an annotation and choosing **Set Numbers Relative to This Annotation**.

- **Numbers on plus strand.** Whether to set the numbers relative to the positive or the negative strand in a nucleotide sequence (only applies to DNA sequences).

- **Follow selection.** When viewing the same sequence in two separate views, "Follow selection" will automatically scroll the view in order to follow a selection made in the other view.

- **Lock numbers.** When you scroll vertically, the position numbers remain visible. (Only possible when the sequence is not wrapped.)

- **Lock labels.** When you scroll horizontally, the label of the sequence remains visible.

- **Sequence label.** Defines the label to the left of the sequence.
  - Name (this is the default information to be shown).
  - Accession (sequences downloaded from databases like GenBank have an accession number).
  - Latin name.
  - Latin name (accession).
  - Common name.
  - Common name (accession).

Annotation Layout and Annotation Types

See section 10.3.1.

Restriction sites

See section 10.1.2.

Motifs

See section 14.7.1.

Residue coloring

These preferences make it possible to color both the residue letter and set a background color for the residue.

- **Non-standard residues.** For nucleotide sequences this will color the residues that are not C, G, A, T or U. For amino acids only B, Z, and X are colored as non-standard residues.
  - **Foreground color.** Sets the color of the letter. Click the color box to change the color.
  - **Background color.** Sets the background color of the residues. Click the color box to change the color.
- **Rasmol colors.** Colors the residues according to the Rasmol color scheme. See http://www.openrasmol.org/doc/rasmol.html
  - **Foreground color.** Sets the color of the letter. Click the color box to change the color.
  - **Background color.** Sets the background color of the residues. Click the color box to change the color.
- **Polarity colors (only protein).** Colors the residues according to the polarity of amino acids.
  - **Foreground color.** Sets the color of the letter. Click the color box to change the color.
  - **Background color.** Sets the background color of the residues. Click the color box to change the color.
- **Trace colors (only DNA).** Colors the residues according to the color conventions of chromatogram traces: A=green, C=blue, G=black, and T=red.
  - **Foreground color.** Sets the color of the letter.
  - **Background color.** Sets the background color of the residues.

Nucleotide info

These preferences only apply to nucleotide sequences.

- **Translation.** Displays a translation into protein just below the nucleotide sequence. Depending on the zoom level, the amino acids are displayed with three letters or one letter.
  - **Frame.** Determines where to start the translation.
    - **ORF/CDS**. If the sequence is annotated, the translation will follow the CDS or ORF annotations. If annotations overlap, only one translation will be shown. If only one annotation is visible, the Workbench will attempt to use this annotation to mark the start and stop for the translation. In cases where this is not possible, the first annotation will be used (i.e. the one closest to the 5' end of the sequence).
    - **+1 to -1.** Select one of the six reading frames.

        - **Selection.** This option will only take effect when you make a selection on the sequence. The translation will start from the first nucleotide selected. Making a new selection will automatically display the corresponding translation. Read more about selecting in section 10.1.3.
        - **All.** Select all reading frames at once. The translations will be displayed on top of each other.
    - **Table.** The translation table to use in the translation. For more about translation tables, see section 15.5.
    - **Only AUG start codons.** For most genetic codes, a number of codons can be start codons. Selecting this option only colors the AUG codons green.
    - **Single letter codes.** Choose to represent the amino acids with a single letter instead of three letters.

- **Trace data.** See section 18.1.

- **Quality scores.** For sequencing data containing quality scores, the quality score information can be displayed along the sequence.

    - **Show as probabilities.** Converts quality scores to error probabilities on a 0-1 scale, i.e. not log-transformed.
    - **Foreground color.** Colors the letter using a gradient, where the left side color is used for low quality and the right side color is used for high quality. The sliders just above the gradient color box can be dragged to highlight relevant levels. The colors can be changed by clicking the box. This will show a list of gradients to choose from.
    - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
    - **Graph.** The quality score is displayed on a graph (Learn how to export the data behind the graph in section 7.4).
        - **Height.** Specifies the height of the graph.
        - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
        - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

- **G/C content.** Calculates the G/C content of a part of the sequence and shows it as a gradient of colors or as a graph below the sequence.

    - **Window length.** Determines the length of the part of the sequence to calculate. A window length of 9 will calculate the G/C content for the nucleotide in question plus the 4 nucleotides to the left and the 4 nucleotides to the right. A narrow window will focus on small fluctuations in the G/C content level, whereas a wider window will show fluctuations between larger parts of the sequence.
    - **Foreground color.** Colors the letter using a gradient, where the left side color is used for low levels of G/C content and the right side color is used for high levels of G/C content. The sliders just above the gradient color box can be dragged to highlight relevant levels of G/C content. The colors can be changed by clicking the box. This will show a list of gradients to choose from.

- **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
- **Graph.** The G/C content level is displayed on a graph (Learn how to export the data behind the graph in section 7.4).
  - **Height.** Specifies the height of the graph.
  - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
  - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

Protein info

These preferences only apply to proteins. The first nine items are different hydrophobicity scales and are described in section 16.5.2.

- **Kyte-Doolittle.** The Kyte-Doolittle scale is widely used for detecting hydrophobic regions in proteins. Regions with a positive value are hydrophobic. This scale can be used for identifying both surface-exposed regions as well as transmembrane regions, depending on the window size used. Short window sizes of 5-7 generally work well for predicting putative surface-exposed regions. Large window sizes of 19-21 are well suited for finding transmembrane domains if the values calculated are above 1.6 [Kyte and Doolittle, 1982]. These values should be used as a rule of thumb and deviations from the rule may occur.

- **Cornette.** Cornette *et al.* computed an optimal hydrophobicity scale based on 28 published scales [Cornette et al., 1987]. This optimized scale is also suitable for prediction of alpha-helices in proteins.

- **Engelman.** The Engelman hydrophobicity scale, also known as the GES-scale, is another scale which can be used for prediction of protein hydrophobicity [Engelman et al., 1986]. As the Kyte-Doolittle scale, this scale is useful for predicting transmembrane regions in proteins.

- **Eisenberg.** The Eisenberg scale is a normalized consensus hydrophobicity scale which shares many features with the other hydrophobicity scales [Eisenberg et al., 1984].

- **Rose.** The hydrophobicity scale by Rose *et al.* is correlated to the average area of buried amino acids in globular proteins [Rose et al., 1985]. This results in a scale which is not showing the helices of a protein, but rather the surface accessibility.

- **Janin.** This scale also provides information about the accessible and buried amino acid residues of globular proteins [Janin, 1979].

- **Hopp-Woods.** Hopp and Woods developed their hydrophobicity scale for identification of potentially antigenic sites in proteins. This scale is basically a hydrophilic index where apolar residues have been assigned negative values. Antigenic sites are likely to be predicted when using a window size of 7 [Hopp and Woods, 1983].

- **Welling.** [Welling et al., 1985] Welling et al. used information on the relative occurrence of amino acids in antigenic regions to make a scale which is useful for prediction of antigenic regions. This method is better than the Hopp-Woods scale of hydrophobicity which is also used to identify antigenic regions.

- **Kolaskar-Tongaonkar.** A semi-empirical method for prediction of antigenic regions has been developed [Kolaskar and Tongaonkar, 1990]. This method also includes information of surface accessibility and flexibility and at the time of publication the method was able to predict antigenic determinants with an accuracy of 75%.
- **Surface Probability.** Display of surface probability based on the algorithm by [Emini et al., 1985]. This algorithm has been used to identify antigenic determinants on the surface of proteins.
- **Chain Flexibility.** Display of backbone chain flexibility based on the algorithm by [Karplus and Schulz, 1985]. It is known that chain flexibility is an indication of a putative antigenic determinant.

Find

The Find function can also be invoked by pressing Ctrl + Shift + F (⌘ + Shift + F on Mac).

The Find function can be used for searching the sequence. Clicking the find button will search for the first occurrence of the search term. Clicking the find button again will find the next occurrence and so on. If the search string is found, the corresponding part of the sequence will be selected.

- **Search term.** Enter the text to search for. The search function does not discriminate between lower and upper case characters.
- **Sequence search.** Search the nucleotides or amino acids. For amino acids, the single letter abbreviations should be used for searching. The sequence search also has a set of advanced search parameters:
  - Include negative strand. This will search on the negative strand as well.
  - Treat ambiguous characters as wildcards in search term. If you search for e.g. ATN, you will find both ATG and ATC. If you wish to find literally exact matches for ATN (i.e. only find ATN - not ATG), this option should not be selected.
  - Treat ambiguous characters as wildcards in sequence. If you search for e.g. ATG, you will find both ATG and ATN. If you have large regions of Ns, this option should not be selected.

  Note that if you enter a position instead of a sequence, it will automatically switch to position search.
- **Annotation search.** Searches the annotations on the sequence. The search is performed both on the labels of the annotations, but also on the text appearing in the tooltip that you see when you keep the mouse cursor fixed. If the search term is found, the part of the sequence corresponding to the matching annotation is selected. Below this option you can choose to search for translations as well. Sequences annotated with coding regions often have the translation specified which can lead to undesired results.
- **Position search.** Finds a specific position on the sequence. In order to find an interval, e.g. from position 500 to 570, enter "500..570" in the search field. This will make a selection from position 500 to 570 (both included). Notice the two periods (..) between the start an end number (see section 10.3.2). You can also enter positions separated by commas or dots (like 123,345 - in this case the comma will just be ignored)

- **Include negative strand.** When searching the sequence for nucleotides or amino acids, you can search on both strands.
- **Name search.** Searches for sequence names. This is useful for searching sequence lists, mapping results and BLAST results.

This concludes the description of the **View Preferences**. Next, the options for selecting and editing sequences are described.

Text format

These preferences allow you to adjust the format of all the text in the view (both residue letters, sequence name and translations if they are shown).

- **Text size.** Five different sizes.
- **Font.** Shows a list of Fonts available on your computer.
- **Bold residues.** Makes the residues bold.

10.1.2 Restriction sites in the Side Panel

Please see section 21.3.1.

10.1.3 Selecting parts of the sequence

You can select parts of a sequence:

> **Click Selection ( ) in Toolbar | Press and hold down the mouse button on the sequence where you want the selection to start | move the mouse to the end of the selection while holding the button | release the mouse button**

Alternatively, you can search for a specific interval using the find function described above.

If you have made a selection and wish to adjust it:

> **drag the edge of the selection (you can see the mouse cursor change to a horizontal arrow**
>
> or **press and hold the Shift key while using the right and left arrow keys to adjust the right side of the selection.**

If you wish to select the entire sequence:

> **double-click the sequence name to the left**

Selecting several parts at the same time (multiselect)

You can select several parts of sequence by holding down the **Ctrl** button while making selections. Holding down the **Shift** button lets you extend or reduce an existing selection to the position you clicked.

To select a part of a sequence covered by an annotation:

right-click the annotation | Select annotation

or **double-click the annotation**

To select a fragment between two restriction sites that are shown on the sequence:

double-click the sequence between the two restriction sites

(Read more about restriction sites in section 10.1.2.)

Open a selection in a new view

A selection can be opened in a new view and saved as a new sequence:

right-click the selection | Open selection in New View ()

This opens the annotated part of the sequence in a new view. The new sequence can be saved by dragging the tab of the sequence view into the **Navigation Area**.

The process described above is also the way to manually translate coding parts of sequences (CDS) into protein. You simply translate the new sequence into protein. This is done by:

right-click the tab of the new sequence | Toolbox | Nucleotide Analyses () | Translate to Protein ()

A selection can also be copied to the clipboard and pasted into another program:

make a selection | Ctrl + C (⌘ + C on Mac)

Note! The annotations covering the selection will not be copied.

A selection of a sequence can be edited as described in the following section.

10.1.4 Editing the sequence

When you make a selection, it can be edited by:

right-click the selection | Edit Selection ()

A dialog appears displaying the sequence. You can add, remove or change the text and click **OK**. The original selected part of the sequence is now replaced by the sequence entered in the dialog. This dialog also allows you to paste text into the sequence using Ctrl + V (⌘ + V on Mac).

If you delete the text in the dialog and press **OK**, the selected text on the sequence will also be deleted. Another way to delete a part of the sequence is to:

right-click the selection | Delete Selection ()

If you wish to only correct only one residue, this is possible by simply making the selection only cover one residue and then type the new residue. Another way to edit the sequence is by inserting a restriction site. See section 21.1.4.

10.1.5 Sequence region types

The various annotations on sequences cover parts of the sequence. Some cover an interval, some cover intervals with unknown endpoints, some cover more than one interval etc. In the following, all of these will be referred to as *regions*. Regions are generally illustrated by markings (often arrows) on the sequences. An arrow pointing to the right indicates that the corresponding region is located on the positive strand of the sequence. Figure 10.2 is an example of three regions with separate colors.

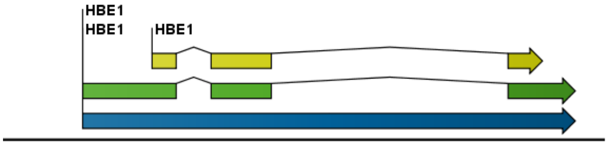


Figure 10.2: *Three regions on a human beta globin DNA sequence (HUMHBB).*

Figure 10.3 shows an artificial sequence with all the different kinds of regions.

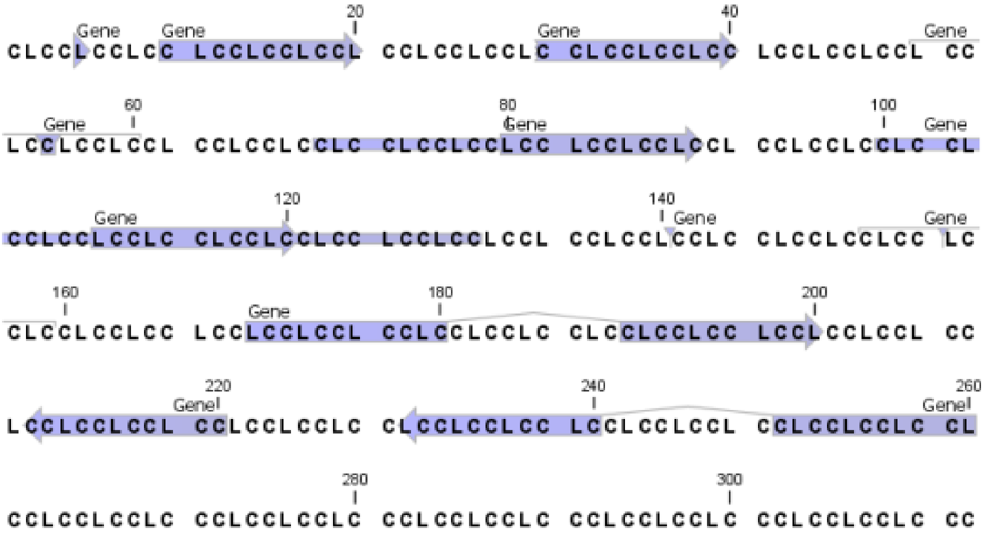

20
40
Gene
Gene
Gene
Gene
CLCCLCCLCC LCCLCCLCCL CCLCCLCCLC CLCCLCCLCC LCCLCCLCCL CC
60
80
100
Gene
Gene
Gene
LCCLCCLCCL CCLCCLCCLC CLCCLCCLCC LCCLCCLCCL CCLCCLCCLC CL
120
140
Gene
Gene
Gene
CCLCCLCCLC CLCCLCCLCC LCCLCCLCCL CCLCCLCCLC CLCCLCCLCC LC
160
180
200
Gene
CLCCLCCLCC LCCLCCLCCL CCLCCLCCLC CLCCLCCLCC LCCLCCLCCL CC
220
240
260
Gene
Gene
LCCLCCLCCL CCLCCLCCLC CLCCLCCLCC LCCLCCLCCL CCLCCLCCLC CL
280
300
CCLCCLCCLC CCLCCLCCLC CCLCCLCCLC CCLCCLCCLC CCLCCLCCLC CC

Figure 10.3: *Region #1: A single residue, Region #2: A range of residues including both endpoints, Region #3: A range of residues starting somewhere before 30 and continuing up to and including 40, Region #4: A single residue somewhere between 50 and 60 inclusive, Region #5: A range of residues beginning somewhere between 70 and 80 inclusive and ending at 90 inclusive, Region #6: A range of residues beginning somewhere between 100 and 110 inclusive and ending somewhere between 120 and 130 inclusive, Region #7: A site between residues 140 and 141, Region #8: A site between two residues somewhere between 150 and 160 inclusive, Region #9: A region that covers ranges from 170 to 180 inclusive and 190 to 200 inclusive, Region #10: A region on negative strand that covers ranges from 210 to 220 inclusive, Region #11: A region on negative strand that covers ranges from 230 to 240 inclusive and 250 to 260 inclusive.*

10.2 Circular DNA

A sequence can be shown as a circular molecule:

select a sequence in the Navigation Area | Show in the Toolbar | As Circular (○)

or **If the sequence is already open | Click Show As Circular (○) at the lower left part of the view**

This will open a view of the molecule similar to the one in figure 10.4.

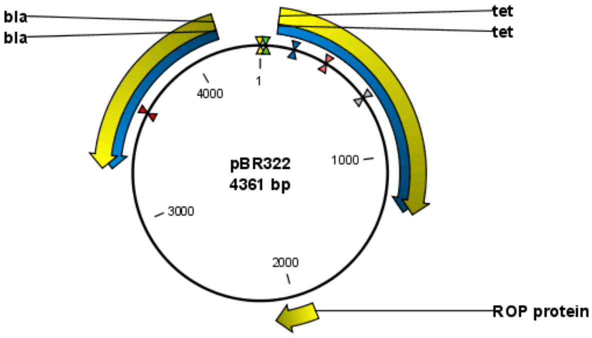


Figure 10.4: *A molecule shown in a circular view.*

This view of the sequence shares some of the properties of the linear view of sequences as described in section 10.1, but there are some differences. The similarities and differences are listed below:

- **Similarities**:
  - The editing options.
  - Options for adding, editing and removing annotations.
  - **Restriction Sites**, **Annotation Types**, **Find** and **Text Format** preferences groups.
- **Differences**:
  - In the **Sequence Layout** preferences, only the following options are available in the circular view: **Numbers on plus strand**, **Numbers on sequence** and **Sequence label**.
  - You cannot zoom in to see the residues in the circular molecule. If you wish to see these details, split the view with a linear view of the sequence
  - In the **Annotation Layout**, you also have the option of showing the labels as **Stacked**. This means that there are no overlapping labels and that all labels of both annotations and restriction sites are adjusted along the left and right edges of the view.

10.2.1 Using split views to see details of the circular molecule

In order to see the nucleotides of a circular molecule you can open a new view displaying a circular view of the molecule:

> **Press and hold the Ctrl button (⌘ on Mac) | click Show Sequence (ACT) at the bottom of the view**

This will open a linear view of the sequence below the circular view. When you zoom in on the linear view you can see the residues as shown in figure 10.5.

Note! If you make a selection in one of the views, the other view will also make the corresponding selection, providing an easy way for you to focus on the same region in both views.

10.2.2 Mark molecule as circular and specify starting point

You can mark a DNA molecule as circular by right-clicking its name in either the sequence view or the circular view. In the right-click menu you can also make a circular molecule linear. A circular molecule displayed in the normal sequence view, will have the sequence ends marked with a ».

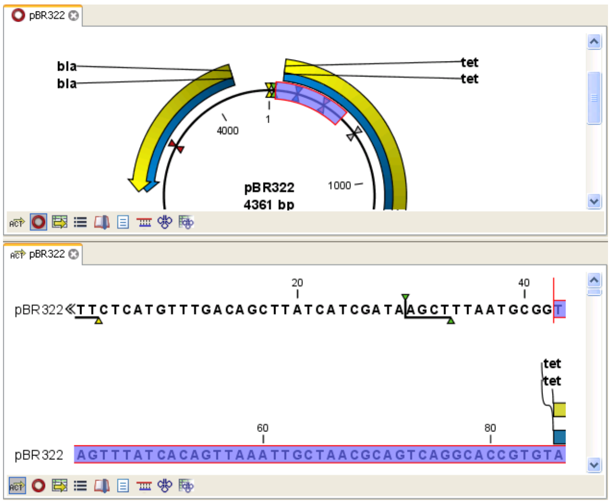

pBR322
bla
bla
tet
tet
4000
1
1000
pBR322
4361 bp
pBR322
20
40
pBR322《TTCTCATGTTTGACAGCTTATCATCGATAAGCTTTAATGCGGT
tet
tet
60
80
pBR322 AGTTTATCACAGTTAAATTGCTAACGCAGTCAGGCACCGTGTA

Figure 10.5: *Two views showing the same sequence. The bottom view is zoomed in.*

The starting point of a circular sequence can be changed by:

> **make a selection starting at the position that you want to be the new starting point | right-click the selection | Move Starting Point to Selection Start**

Note! This can only be done for sequence that have been marked as circular.

10.3 Working with annotations

Annotations provide information about specific regions of a sequence. A typical example is the annotation of a gene on a genomic DNA sequence.

Annotations derive from different sources:

- Sequences downloaded from databases like GenBank are annotated.
- In some of the data formats that can be imported into *CLC Genomics Workbench*, sequences can have annotations (GenBank, EMBL and Swiss-Prot format).
- The result of a number of analyses in *CLC Genomics Workbench* are annotations on the sequence (e.g. finding open reading frames and restriction map analysis).
- You can manually add annotations to a sequence (described in the section 10.3.2).

Note! Annotations are included if you export the sequence in GenBank, Swiss-Prot, EMBL or CLC format. When exporting in other formats, annotations are not preserved in the exported file.

10.3.1 Viewing annotations

Annotations can be viewed in a number of different ways:

- As arrows or boxes in the sequence views:
  - Linear and circular view of sequences (ACT) / (○).
  - Alignments (☰).

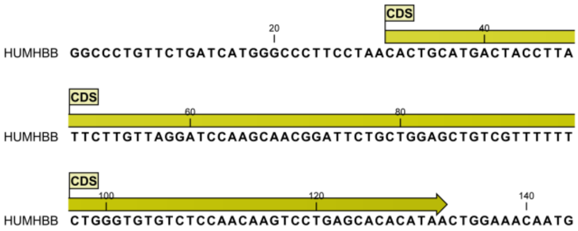
  - Graphical view of sequence lists ().
  - BLAST views (only the query sequence at the top can have annotations) ().
  - Cloning editor ().
  - Primer designer (both for single sequences and alignments) () / ().
  - Contig/mapping view ().
- In the table of annotations ().
- In the text view of sequences ()

In the following sections, these view options will be described in more detail.

In all the views except the text view (), annotations can be added, modified and deleted. This is described in the following sections.

View Annotations in sequence views

Figure 10.6 shows an annotation displayed on a sequence.



```
HUMHBB GGCCCTGTTCTGATCATGGGCCCTTCCTAACACTGCATGACTACCTTA
HUMHBB TTCTTGTTAGGATCCAAGCAACGGATTCTGCTGGAGCTGTCGTTTTTT
HUMHBB CTGGGTGTGTCTCCAACAAGTCCTGAGCACACATAACTGGAAACAATG
```

Figure 10.6: *An annotation showing a coding region on a genomic dna sequence.*

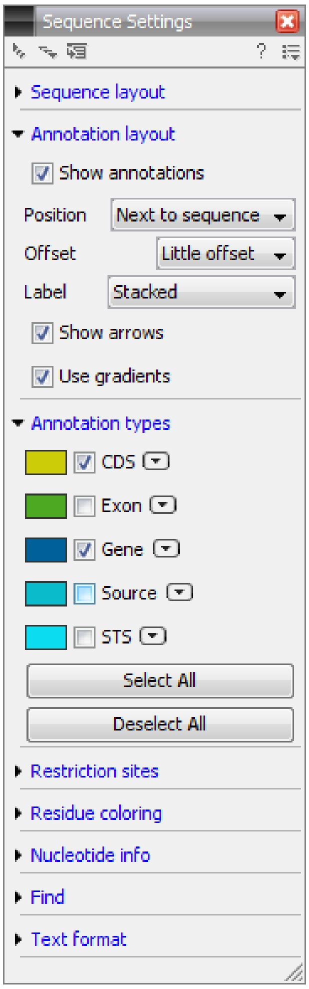
The various sequence views listed in section 10.3.1 have different default settings for showing annotations. However, they all have two groups in the **Side Panel** in common:

- **Annotation Layout**
- **Annotation Types**

The two groups are shown in figure 10.7.

In the **Annotation layout** group, you can specify how the annotations should be displayed (notice that there are some minor differences between the different sequence views):

- **Show annotations.** Determines whether the annotations are shown.
- **Position.**
  - **On sequence.** The annotations are placed on the sequence. The residues are visible through the annotations (if you have zoomed in to 100%).
  - **Next to sequence.** The annotations are placed above the sequence.
  - **Separate layer.** The annotations are placed above the sequence and above restriction sites (only applicable for nucleotide sequences).



Sequence Settings

Sequence layout

Annotation layout

☑ Show annotations

Position: Next to sequence

Offset: Little offset

Label: Stacked

☑ Show arrows

☑ Use gradients

Annotation types

🟨 ☑ CDS

🟩 ☐ Exon

🟦 ☑ Gene

🟦 ☐ Source

🟦 ☐ STS

Figure 10.7: *Changing the layout of annotations in the Side Panel.*

- **Offset.** If several annotations cover the same part of a sequence, they can be spread out.
  - **Piled.** The annotations are piled on top of each other. Only the one at front is visible.
  - **Little offset.** The annotations are piled on top of each other, but they have been offset a little.
  - **More offset.** Same as above, but with more spreading.
  - **Most offset.** The annotations are placed above each other with a little space between. This can take up a lot of space on the screen.
- **Label.** The name of the annotation can shown as a label. Additional information about the sequence is shown if you place the mouse cursor on the annotation and keep it still.
  - **No labels.** No labels are displayed.
  - **On annotation.** The labels are displayed in the annotation's box.
  - **Over annotation.** The labels are displayed above the annotations.
  - **Before annotation.** The labels are placed just to the left of the annotation.
  - **Flag.** The labels are displayed as flags at the beginning of the annotation.
  - **Stacked.** The labels are offset so that the text of all labels is visible. This means that there is varying distance between each sequence line to make room for the labels.

- **Show arrows.** Displays the end of the annotation as an arrow. This can be useful to see the orientation of the annotation (for DNA sequences). Annotations on the negative strand will have an arrow pointing to the left.
- **Use gradients.** Fills the boxes with gradient color.

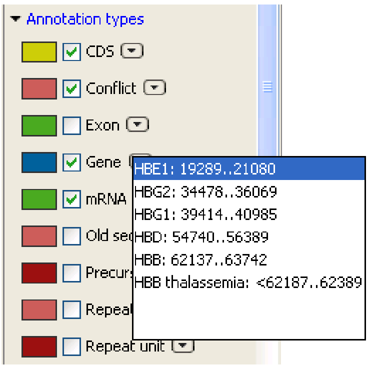
In the **Annotation Types** group, you can choose which kinds of annotations that should be displayed. This group lists all the types of annotations that are attached to the sequence(s) in the view. For sequences with many annotations, it can be easier to get an overview if you deselect the annotation types that are not relevant.

Unchecking the checkboxes in the **Annotation Layout** will not remove this type of annotations them from the sequence - it will just hide them from the view.

Besides selecting which types of annotations that should be displayed, the **Annotation Types** group is also used to change the color of the annotations on the sequence. Click the colored square next to the relevant annotation type to change the color.

This will display a dialog with three tabs: Swatches, HSB, and RGB. They represent three different ways of specifying colors. Apply your settings and click **OK**. When you click **OK**, the color settings cannot be reset. The **Reset** function only works for changes made before pressing **OK**.

Furthermore, the **Annotation Types** can be used to easily browse the annotations by clicking the small button (▾) next to the type. This will display a list of the annotations of that type (see figure 10.8).



☑ CDS
☑ Conflict
□ Exon
☑ Gene
☑ mRNA
□ Old se
□ Precur
□ Repeat
□ Repeat unit

HBE1: 19289..21080
HBG2: 34478..36069
HBG1: 39414..40985
HBD: 54740..56389
HBB: 62137..63742
HBB thalassemia: <62187..62389

Figure 10.8: *Browsing the gene annotations on a sequence.*

Clicking an annotation in the list will select this region on the sequence. In this way, you can quickly find a specific annotation on a long sequence.

View Annotations in a table

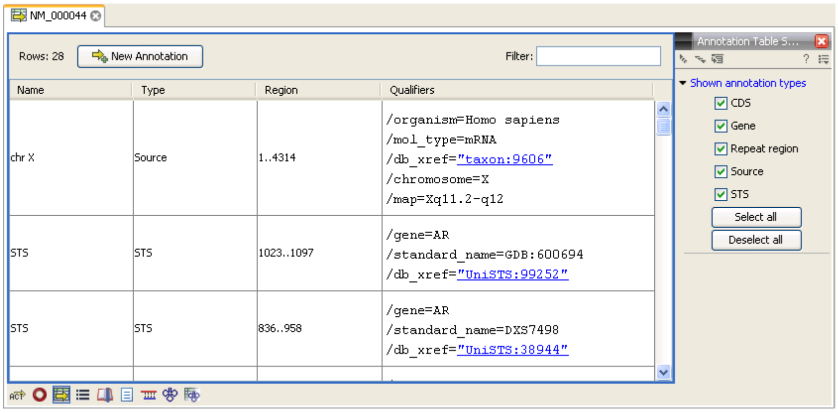
Annotations can also be viewed in a table:

> **select the sequence in the Navigation Area | Show ( ) | Annotation Table ( )**

or **If the sequence is already open | Click Show Annotation Table ( ) at the lower left part of the view**

This will open a view similar to the one in figure 10.9).

In the **Side Panel** you can show or hide individual annotation types in the table. E.g. if you



Rows: 28

Filter:

Name	Type	Region	Qualifiers
chr X	Source	1..4314	/organism=Homo sapiens /mol_type=mRNA /db_xref="taxon:9606" /chromosome=X /map=Xq11.2-q12
STS	STS	1023..1097	/gene=AR /standard_name=GDB:600694 /db_xref="UniSTS:99252"
STS	STS	836..958	/gene=AR /standard_name=DXS7498 /db_xref="UniSTS:38944"

Shown annotation types

☑ CDS

☑ Gene

☑ Repeat region

☑ Source

☑ STS

Figure 10.9: *A table showing annotations on the sequence.*

only wish to see "gene" annotations, de-select the other annotation types so that only "gene" is selected.

Each row in the table is an annotation which is represented with the following information:

- **Name.**
- **Type.**
- **Region.**
- **Qualifiers.**

The Name, Type and Region for each annotation can be edited simply by double-clicking, typing the change directly, and pressing **Enter**.

This information corresponds to the information in the dialog when you edit and add annotations (see section 10.3.2).

You can benefit from this table in several ways:

- It provides an intelligible overview of all the annotations on the sequence.
- You can use the filter at the top to search the annotations. Type e.g. "UCP" into the filter and you will find all annotations which have "UCP" in either the name, the type, the region or the qualifiers. Combined with showing or hiding the annotation types in the **Side Panel**, this makes it easy to find annotations or a subset of annotations.
- You can copy and paste annotations, e.g. from one sequence to another.
- If you wish to edit many annotations consecutively, the double-click editing makes this very fast (see section 10.3.2).

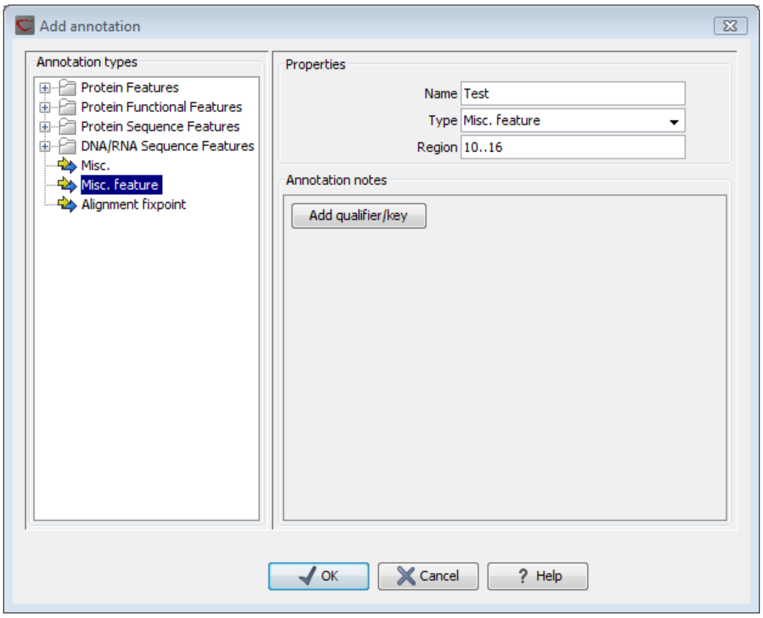
10.3.2 Adding annotations

Adding annotations to a sequence can be done in two ways:

open the sequence in a sequence view (double-click in the Navigation Area) | make a selection covering the part of the sequence you want to annotate[1] **| right-click the selection | Add Annotation (**➯**)**

or **select the sequence in the Navigation Area | Show () | Annotations () | Add Annotation ()**

This will display a dialog like the one in figure 10.10.



Name: Test
Type: Misc. feature
Region: 10..16

Figure 10.10: *The Add Annotation dialog.*

The left-hand part of the dialog lists a number of **Annotation types**. When you have selected an annotation type, it appears in **Type** to the right. You can also select an annotation directly in this list. Choosing an annotation type is mandatory. If you wish to use an annotation type which is not present in the list, simply enter this type into the **Type** field [2].

The right-hand part of the dialog contains the following text fields:

- **Name.** The name of the annotation which can be shown on the label in the sequence views. (Whether the name is actually shown depends on the **Annotation Layout** preferences, see section 10.3.1).
- **Type.** Reflects the left-hand part of the dialog as described above. You can also choose directly in this list or type your own annotation type.
- **Region.** If you have already made a selection, this field will show the positions of the selection. You can modify the region further using the conventions of DDBJ, EMBL and GenBank. The following are examples of how to use the syntax (based on `http://www.ncbi.nlm.nih.gov/collab/FT/`):
  - **467**. Points to a single residue in the presented sequence.
  - **340..565**. Points to a continuous range of residues bounded by and including the starting and ending residues.
  - **<345..500**. Indicates that the exact lower boundary point of a region is unknown. The location begins at some residue previous to the first residue specified (which is not

[2] Note that your own annotation types will be converted to "unsure" when exporting in GenBank format. As long as you use the sequence in CLC format, you own annotation type will be preserved

necessarily contained in the presented sequence) and continues up to and including the ending residue.

  - **<1..888**. The region starts before the first sequenced residue and continues up to and including residue 888.
  - **1..>888**. The region starts at the first sequenced residue and continues beyond residue 888.
  - **(102.110)**. Indicates that the exact location is unknown, but that it is one of the residues between residues 102 and 110, inclusive.
  - **123ˆ124**. Points to a site between residues 123 and 124.
  - **join(12..78,134..202)**. Regions 12 to 78 and 134 to 202 should be joined to form one contiguous sequence.
  - **complement(34..126)** Start at the residue complementary to 126 and finish at the residue complementary to residue 34 (the region is on the strand complementary to the presented strand).
  - **complement(join(2691..4571,4918..5163))**. Joins regions 2691 to 4571 and 4918 to 5163, then complements the joined segments (the region is on the strand complementary to the presented strand).
  - **join(complement(4918..5163),complement(2691..4571))**. Complements regions 4918 to 5163 and 2691 to 4571, then joins the complemented segments (the region is on the strand complementary to the presented strand).

- **Annotations.** In this field, you can add more information about the annotation like comments and links. Click the **Add qualifier/key** button to enter information. Select a qualifier which describes the kind of information you wish to add. If an appropriate qualifier is not present in the list, you can type your own qualifier. The pre-defined qualifiers are derived from the GenBank format. You can add as many qualifier/key lines as you wish by clicking the button. Redundant lines can be removed by clicking the delete icon (☒). The information entered on these lines is shown in the annotation table (see section 10.3.1) and in the yellow box which appears when you place the mouse cursor on the annotation. If you write a hyperlink in the **Key** text field, like e.g. "www.clcbio.com", it will be recognized as a hyperlink. Clicking the link in the annotation table will open a web browser.

Click **OK** to add the annotation.

Note! The annotation will be included if you export the sequence in GenBank, Swiss-Prot or CLC format. When exporting in other formats, annotations are not preserved in the exported file.

10.3.3 Edit annotations

To edit an existing annotation from within a sequence view:

right-click the annotation | Edit Annotation (✎)

This will show the same dialog as in figure 10.10, with the exception that some of the fields are filled out depending on how much information the annotation contains.

There is another way of quickly editing annotations which is particularly useful when you wish to edit several annotations.

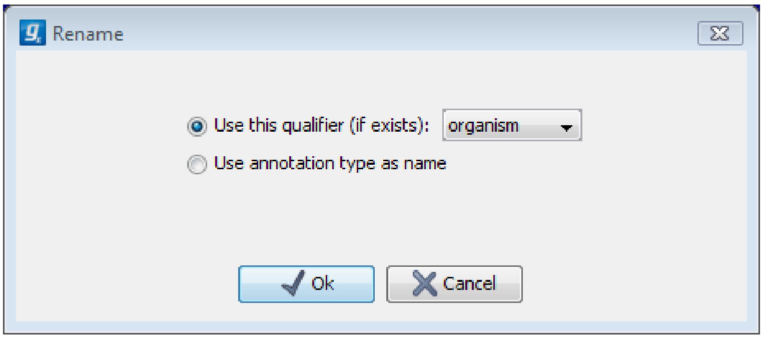
To edit the information, simply double-click and you will be able to edit e.g. the name or the annotation type. If you wish to edit the qualifiers and double-click in this column, you will see the dialog for editing annotations.

Advanced editing of annotations

Sometimes you end up with annotations which do not have a meaningful name. In that case there is an advanced batch rename functionality:

> **Open the Annotation Table () | select the annotations that you want to rename | right-click the selection | Advanced Rename**

This will bring up the dialog shown in figure 10.11.



Rename

● Use this qualifier (if exists): organism

○ Use annotation type as name

Figure 10.11: *The Advanced Rename dialog.*

In this dialog, you have two options:

- **Use this qualifier.** Use one of the qualifiers as name. A list of all qualifiers of all the selected annotations is shown. Note that if one of the annotations do not have the qualifier you have chosen, it will not be renamed. If an annotation has multiple qualifiers of the same type, the first is used for naming.
- **Use annotation type as name.** The annotation's type will be used as name (e.g. if you have an annotation of type "Promoter", it will get "Promoter" as its name by using this option).

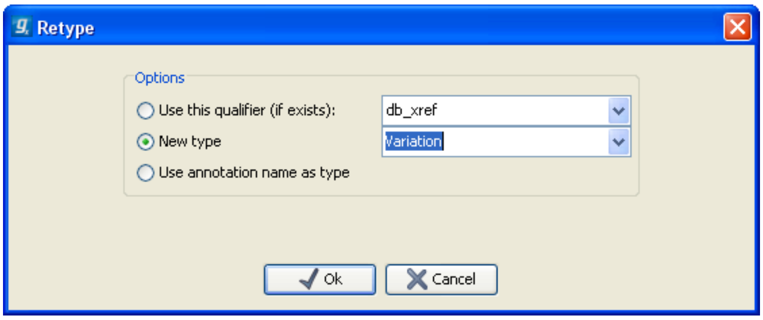
A similar functionality is available for batch re-typing annotations is available in the right-click menu as well, in case your annotations are not typed correctly:

> **Open the Annotation Table () | select the annotations that you want to retype | right-click the selection | Advanced Retype**

This will bring up the dialog shown in figure 10.12.

In this dialog, you have two options:

- **Use this qualifier.** Use one of the qualifiers as type. A list of all qualifiers of all the selected annotations is shown. Note that if one of the annotations do not have the qualifier you have chosen, it will not be retyped. If an annotation has multiple qualifiers of the same type, the first is used for the new type.
- **New type**. You can select from a list of all the pre-defined types as well as enter your own annotation type. All the selected annotations will then get this type.



Options

○ Use this qualifier (if exists): db_xref

● New type Variation

○ Use annotation name as type

Figure 10.12: *The Advanced Retype dialog.*

- **Use annotation name as type.** The annotation's name will be used as type (e.g. if you have an annotation named "Promoter", it will get "Promoter" as its type by using this option).

10.3.4 Removing annotations

Annotations can be hidden using the **Annotation Types** preferences in the **Side Panel** to the right of the view (see section 10.3.1). In order to completely remove the annotation:

right-click the annotation | Delete | Delete Annotation ()

If you want to remove all annotations of one type:

right-click an annotation of the type you want to remove | Delete | Delete Annotations of Type "type"

If you want to remove all annotations from a sequence:

right-click an annotation | Delete | Delete All Annotations

The removal of annotations can be undone using Ctrl + Z or Undo () in the Toolbar.

If you have more sequences (e.g. in a sequence list, alignment or contig), you have two additional options:

right-click an annotation | Delete | Delete All Annotations from All Sequences

right-click an annotation | Delete | Delete Annotations of Type "type" from All Sequences

10.4 Element information

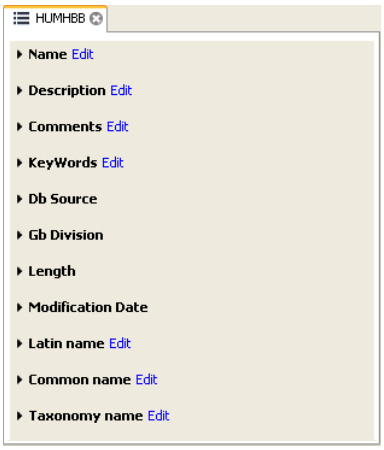
The normal view of a sequence (by double-clicking) shows the annotations as boxes along the sequence, but often there is more information available about sequences. This information is available through the **Element info** view.

To view the sequence information:

select a sequence in the Navigation Area | Show () in the Toolbar | Element info ()

This will display a view similar to fig 10.13.

All the lines in the view are headings, and the corresponding text can be shown by clicking the text.



HUMHBB

- Name Edit
- Description Edit
- Comments Edit
- KeyWords Edit
- Db Source
- Gb Division
- Length
- Modification Date
- Latin name Edit
- Common name Edit
- Taxonomy name Edit

Figure 10.13: *The initial display of sequence info for the HUMHBB DNA sequence from the Example data.*

- **Name.** The name of the sequence which is also shown in sequence views and in the **Navigation Area**.
- **Description.** A description of the sequence.
- **Comments.** The author's comments about the sequence.
- **Keywords.** Keywords describing the sequence.
- **Db source.** Accession numbers in other databases concerning the same sequence.
- **Gb Division.** Abbreviation of GenBank divisions. See section 3.3 in the GenBank release notes for a full list of GenBank divisions.
- **Length.** The length of the sequence.
- **Modification date.** Modification date from the database. This means that this date does not reflect your own changes to the sequence. See the history (section 8) for information about the latest changes to the sequence after it was downloaded from the database.
- **Organism.** Scientific name of the organism (first line) and taxonomic classification levels (second and subsequent lines).

The information available depends on the origin of the sequence. Sequences downloaded from database like NCBI and UniProt (see section 11) have this information. On the other hand, some sequence formats like fasta format do not contain this information.

Some of the information can be edited by clicking the blue **Edit** text. This means that you can add your own information to sequences that do not derive from databases.

Note that for other kinds of data, the **Element info** will only have **Name** and **Description**.

10.5 View as text

A sequence can be viewed as text without any layout and text formatting. This displays all the information about the sequence in the GenBank file format. To view a sequence as text:

select a sequence in the Navigation Area | Show in the Toolbar | As text

This way it is possible to see background information about e.g. the authors and the origin of DNA and protein sequences. Selections or the entire text of the **Sequence Text View** can be copied and pasted into other programs:

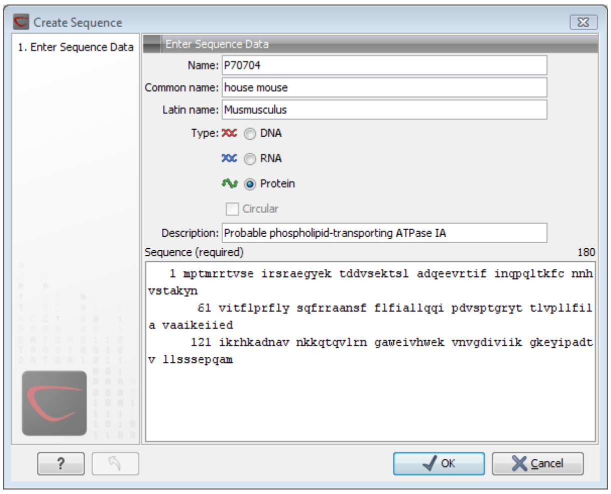
Much of the information is also displayed in the **Sequence info**, where it is easier to get an overview (see section 10.4.)

In the **Side Panel**, you find a search field for searching the text in the view.

10.6 Creating a new sequence

A sequence can either be imported, downloaded from an online database or created in the *CLC Genomics Workbench*. This section explains how to create a new sequence:

New ( ) in the toolbar



Name: P70704

Common name: house mouse

Latin name: Musmusculus

Type: ○ DNA ○ RNA ● Protein

□ Circular

Description: Probable phospholipid-transporting ATPase IA

Sequence (required) 180

```
   1 mptmrrtvse irsraegyek tddvsektsl adqeevrtif inqpqltkfc nnh
vstakyn
      61 vitflprfly sqfrraansf flfiallqqi pdvsptgryt tlvpllfil
a vaaikeiied
     121 ikrhkadnav nkkqtqvlrn gaweivhwek vnvgdiviik gkeyipadt
v llsssepqam
```

Figure 10.14: *Creating a sequence.*

The **Create Sequence** dialog (figure 10.14) reflects the information needed in the GenBank format, but you are free to enter anything into the fields. The following description is a guideline for entering information about a sequence:

- **Name.** The name of the sequence. This is used for saving the sequence.
- **Common name.** A common name for the species.
- **Latin name.** The Latin name for the species.
- **Type.** Select between DNA, RNA and protein.
- **Circular.** Specifies whether the sequence is circular. This will open the sequence in a circular view as default. (applies only to nucleotide sequences).
- **Description.** A description of the sequence.
- **Keywords.** A set of keywords separated by semicolons (;).

- **Comments.** Your own comments to the sequence.
- **Sequence.** Depending on the type chosen, this field accepts nucleotides or amino acids. Spaces and numbers can be entered, but they are ignored when the sequence is created. This allows you to paste (Ctrl + V on Windows and ⌘ + V on Mac) in a sequence directly from a different source, even if the residue numbers are included. Characters that are not part of the IUPAC codes cannot be entered. At the top right corner of the field, the number of residues are counted. The counter does not count spaces or numbers.

Clicking **Finish** opens the sequence. It can be saved by clicking **Save** (💾) or by dragging the tab of the sequence view into the **Navigation Area**.

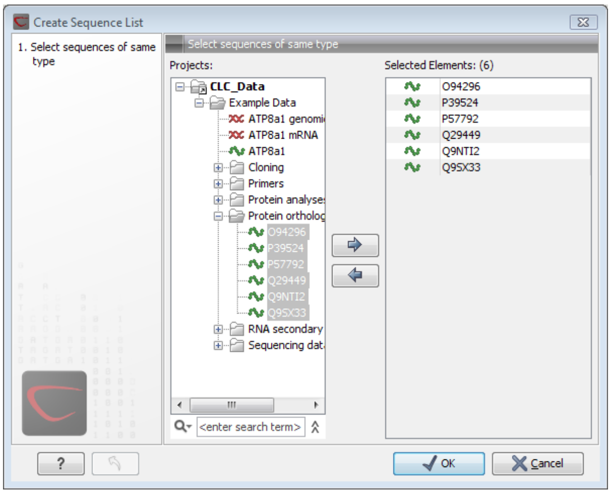
10.7 Sequence Lists

The **Sequence List** shows a number of sequences in a tabular format or it can show the sequences together in a normal sequence view.

Having sequences in a sequence list can help organizing sequence data. The sequence list may originate from an NCBI search (chapter 11.1). Moreover, if a multiple sequence fasta file is imported, it is possible to store the data in a sequences list. A **Sequence List** can also be generated using a dialog, which is described here:

select two or more sequences | **right-click the elements** | **New** | **Sequence List (☰)**

This action opens a **Sequence List** dialog:



O94296
P39524
P57792
Q29449
Q9NTI2
Q9SX33

Figure 10.15: *A Sequence List dialog.*

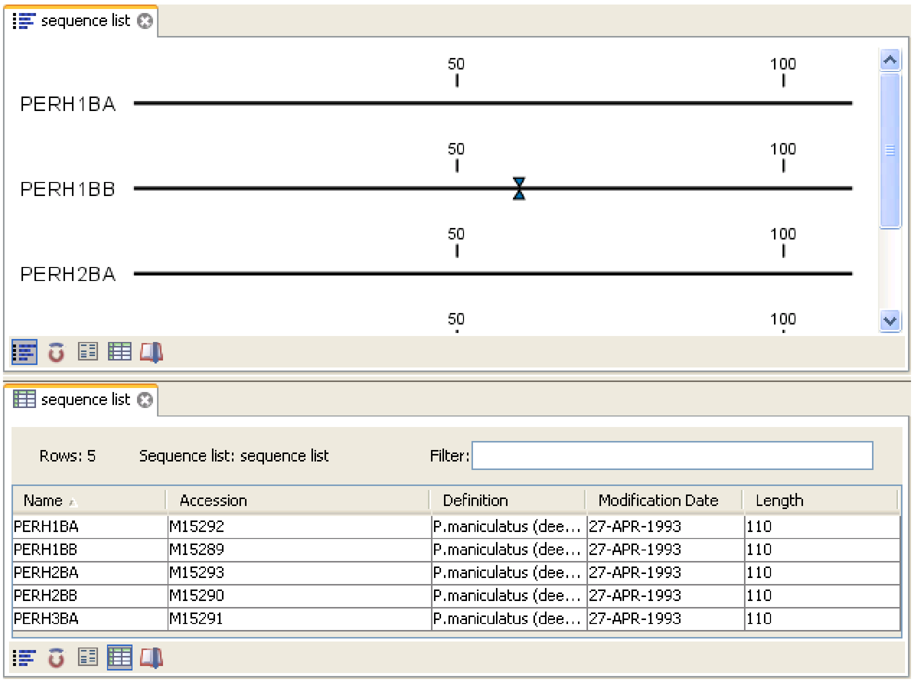
The dialog allows you to select more sequences to include in the list, or to remove already chosen sequences from the list.

Clicking **Finish** opens the sequence list. It can be saved by clicking **Save** (💾) or by dragging the tab of the view into the **Navigation Area**.

Opening a Sequence list is done by:

right-click the sequence list in the Navigation Area | **Show ( )** | **Graphical Sequence List (☰) OR Table (▦)**

The two different views of the same sequence list are shown in split screen in figure 10.16.



Rows: 5 Sequence list: sequence list Filter:

Name	Accession	Definition	Modification Date	Length
PERH1BA	M15292	P.maniculatus (dee...	27-APR-1993	110
PERH1BB	M15289	P.maniculatus (dee...	27-APR-1993	110
PERH2BA	M15293	P.maniculatus (dee...	27-APR-1993	110
PERH2BB	M15290	P.maniculatus (dee...	27-APR-1993	110
PERH3BA	M15291	P.maniculatus (dee...	27-APR-1993	110

Figure 10.16: *A sequence list of two sequences can be viewed in either a table or in a graphical sequence list.*

10.7.1 Graphical view of sequence lists

The graphical view of sequence lists is almost identical to the view of single sequences (see section 10.1). The main difference is that you now can see more than one sequence in the same view.

However, you also have a few extra options for sorting, deleting and adding sequences:

- To add extra sequences to the list, right-click an empty (white) space in the view, and select **Add Sequences**.
- To delete a sequence from the list, right-click the sequence's name and select **Delete Sequence**.
- To sort the sequences in the list, right-click the name of one of the sequences and select **Sort Sequence List by Name** or **Sort Sequence List by Length**.
- To rename a sequence, right-click the name of the sequence and select **Rename Sequence**.

10.7.2 Sequence list table

Each sequence in the table sequence list is displayed with:

- Name.
- Accession.

- Description.
- Modification date.
- Length.

Learn more about tables in section C.

Adding and removing sequences from the list is easy: adding is done by dragging the sequence from another list or from the **Navigation Area** and drop it in the table. To delete sequences, simply select them and press **Delete** ().

You can also create a subset of the sequence list:

select the relevant sequences | **right-click** | **Create New Sequence List**

This will create a new sequence list which only includes the selected sequences.

10.7.3 Extract sequences

It is possible to extract individual sequences from a sequence list in two ways. If the sequence list is opened in the tabular view, it is possible to drag (with the mouse) one or more sequences into the **Navigation Area**. This allows you to extract specific sequences from the entire list. Another option is to extract all sequences found in the list. This can also be done for:

- Alignments ()
- Contigs and read mappings ()
- Read mapping tables ()
- BLAST result ()
- BLAST overview tables ()
- RNA-Seq samples ()
- and of course sequence lists ()

For mappings and BLAST results, the main sequences (i.e. reference/consensus and query sequence) will not be extracted.

To extract the sequences:

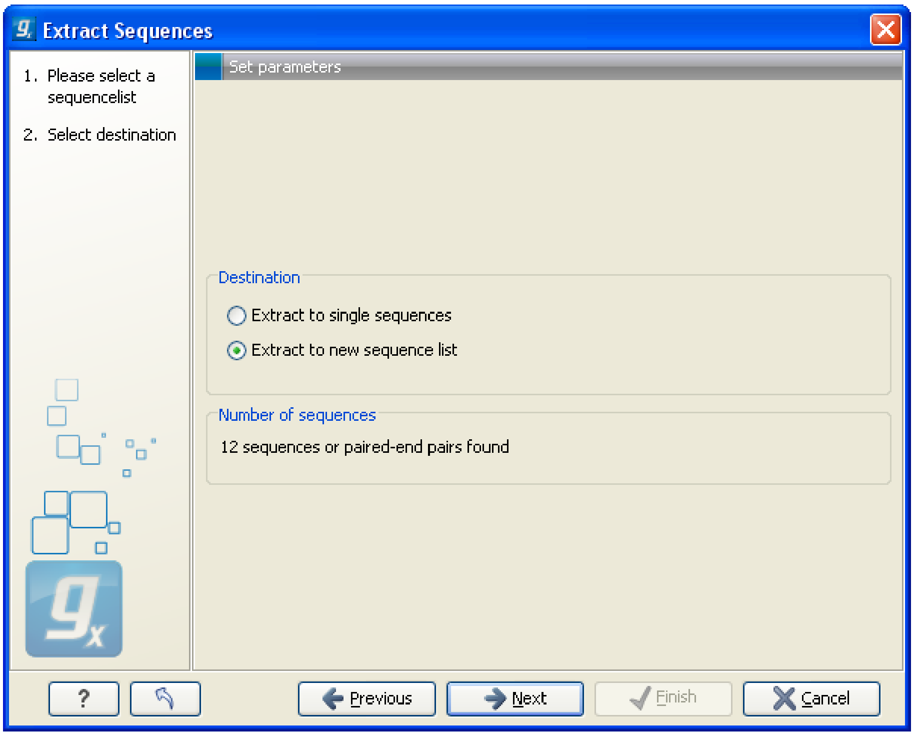
Toolbox | **General Sequence Analyses** () | **Extract Sequences** ()

This will allow you to select the elements that you want to extract sequences from (see the list above). Clicking **Next** displays the dialog shown in 10.17.

Here you can choose whether the extracted sequences should be placed in a new list or extracted as single sequences. For sequence lists, only the last option makes sense, but for alignments, mappings and BLAST results, it would make sense to place the sequences in a list.

Below these options you can see the number of sequences that will be extracted.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.



Destination

○ Extract to single sequences

● Extract to new sequence list

Number of sequences

12 sequences or paired-end pairs found

Figure 10.17: *Choosing whether the extracted sequences should be placed in a new list or as single sequences.*

Chapter 11

Online database search

Contents

CLC Genomics Workbench offers different ways of searching data on the Internet. You must be online when initiating and performing the following searches:

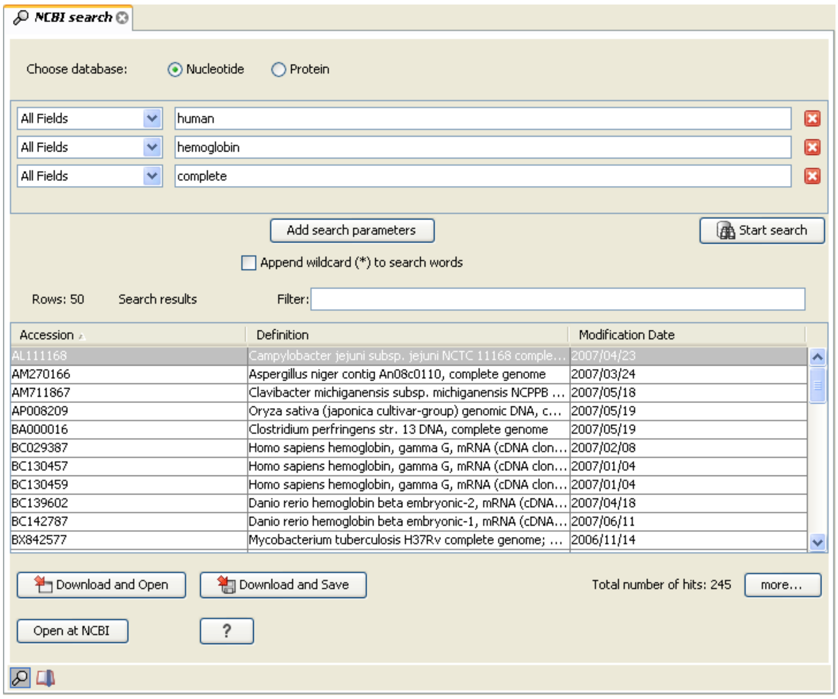
11.1 GenBank search

This section describes searches for sequences in GenBank - the **NCBI Entrez** database. The NCBI search view is opened in this way (figure 11.1):

Search | Search for Sequences at NCBI ( )

or **Ctrl + B (⌘ + B on Mac)**

This opens the following view:



Choose database: ● Nucleotide ○ Protein

All Fields: human

All Fields: hemoglobin

All Fields: complete

□ Append wildcard (*) to search words

Rows: 50 Search results Filter:

Accession	Definition	Modification Date
AL111168	Campylobacter jejuni subsp. jejuni NCTC 11168 comple...	2007/04/23
AM270166	Aspergillus niger contig An08c0110, complete genome	2007/03/24
AM711867	Clavibacter michiganensis subsp. michiganensis NCPPB ...	2007/05/18
AP008209	Oryza sativa (japonica cultivar-group) genomic DNA, c...	2007/05/19
BA000016	Clostridium perfringens str. 13 DNA, complete genome	2007/05/19
BC029387	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/02/08
BC130457	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/01/04
BC130459	Homo sapiens hemoglobin, gamma G, mRNA (cDNA clon...	2007/01/04
BC139602	Danio rerio hemoglobin beta embryonic-2, mRNA (cDNA...	2007/04/18
BC142787	Danio rerio hemoglobin beta embryonic-1, mRNA (cDNA...	2007/06/11
BX842577	Mycobacterium tuberculosis H37Rv complete genome; ...	2006/11/14

Total number of hits: 245

Figure 11.1: *The GenBank search view.*

11.1.1 GenBank search options

Conducting a search in the **NCBI Database** from *CLC Genomics Workbench* corresponds to conducting the search on NCBI's website. When conducting the search from *CLC Genomics Workbench*, the results are available and ready to work with straight away.

You can choose whether you want to search for nucleotide sequences or protein sequences.

As default, *CLC Genomics Workbench* offers one text field where the search parameters can be entered. Click **Add search parameters** to add more parameters to your search.
Note! The search is a "and" search, meaning that when adding search parameters to your search, you search for both (or all) text strings rather than "any" of the text strings.

You can append a wildcard character by checking the checkbox at the bottom. This means that you only have to enter the first part of the search text, e.g. searching for "genom" will find both "genomic" and "genome".

The following parameters can be added to the search:

- **All fields**. Text, searches in all parameters in the NCBI database at the same time.
- **Organism**. Text.
- **Description**. Text.
- **Modified Since**. Between 30 days and 10 years.
- **Gene Location**. Genomic DNA/RNA, Mitochondrion, or Chloroplast.
- **Molecule**. Genomic DNA/RNA, mRNA or rRNA.
- **Sequence Length**. Number for maximum or minimum length of the sequence.

- **Gene Name**. Text.

The search parameters are the most recently used. The **All fields** allows searches in all parameters in the NCBI database at the same time. **All fields** also provide an opportunity to restrict a search to parameters which are not listed in the dialog. E.g. writing `gene[Feature key] AND mouse` in **All fields** generates hits in the GenBank database which contains one or more genes and where 'mouse' appears somewhere in GenBank file. You can also write e.g. `CD9 NOT homo sapiens` in **All fields**.

Note! The 'Feature Key' option is only available in GenBank when searching for nucleotide sequences. For more information about how to use this syntax, see http://www.ncbi.nlm.nih.gov/entrez/query/static/help/helpdoc.html#Writing_Advanced_Search_Statements

When you are satisfied with the parameters you have entered, click **Start search**.

Note! When conducting a search, no files are downloaded. Instead, the program produces a list of links to the files in the NCBI database. This ensures a much faster search.

11.1.2 Handling of GenBank search results

The search result is presented as a list of links to the files in the NCBI database. The **View** displays 50 hits at a time. This can be changed in the **Preferences** (see chapter 5). More hits can be displayed by clicking the **More...** button at the bottom right of the **View**.

Each sequence hit is represented by text in three columns:

- Accession.
- Description.
- Modification date.
- Length.

It is possible to exclude one or more of these columns by adjust the View preferences for the database search view. Furthermore, your changes in the View preferences can be saved. See section 5.6.

Several sequences can be selected, and by clicking the buttons in the bottom of the search view, you can do the following:

- Download and open, doesn't save the sequence.
- Download and save, lets you choose location for saving sequence.
- Open at NCBI, searches the sequence at NCBI's web page.

Double-clicking a hit will download and open the sequence. The hits can also be copied into the **View Area** or the **Navigation Area** from the search results by drag and drop, copy/paste or by using the right-click menu as described below.

Drag and drop from GenBank search results

The sequences from the search results can be opened by dragging them into a position in the **View Area**.

Note! A sequence is not saved until the **View** displaying the sequence is closed. When that happens, a dialog opens: Save changes of sequence x? (Yes or No).

The sequence can also be saved by dragging it into the **Navigation Area**. It is possible to select more sequences and drag all of them into the **Navigation Area** at the same time.

Download GenBank search results using right-click menu

You may also select one or more sequences from the list and download using the right-click menu (see figure 11.2). Choosing **Download and Save** lets you select a folder where the sequences are saved when they are downloaded. Choosing **Download and Open** opens a new view for each of the selected sequences.

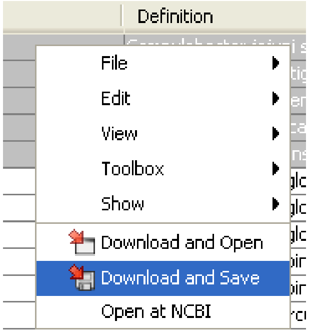


Figure 11.2: *By right-clicking a search result, it is possible to choose how to handle the relevant sequence.*

Copy/paste from GenBank search results

When using copy/paste to bring the search results into the **Navigation Area**, the actual files are downloaded from GenBank.

To copy/paste files into the **Navigation Area**:

> **select one or more of the search results | Ctrl + C (⌘ + C on Mac) | select a folder in the Navigation Area | Ctrl + V**

Note! Search results are downloaded before they are saved. Downloading and saving several files may take some time. However, since the process runs in the background (displayed in the **Status bar**) it is possible to continue other tasks in the program. Like the search process, the download process can be stopped. This is done in the **Toolbox** in the **Processes** tab.

11.1.3 Save GenBank search parameters

The search view can be saved either using dragging the search tab and and dropping it in the **Navigation Area** or by clicking **Save** (🖫). When saving the search, only the parameters are saved - not the results of the search. This is useful if you have a special search that you perform from time to time.

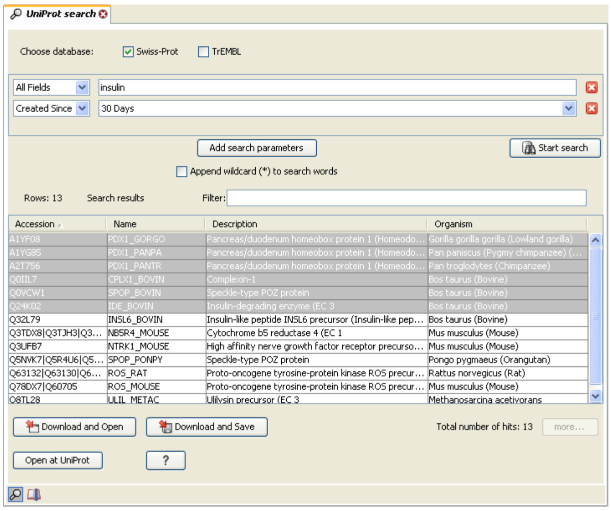
Even if you don't save the search, the next time you open the search view, it will remember the parameters from the last time you did a search.

11.2 UniProt (Swiss-Prot/TrEMBL) search

This section describes searches in UniProt and the handling of search results. UniProt is a global database of protein sequences.

The UniProt search view (figure 11.3) is opened in this way:

Search | Search for Sequences in UniProt ( )



Choose database: ☑ Swiss-Prot □ TrEMBL

All Fields: insulin

Created Since: 30 Days

□ Append wildcard (*) to search words

Rows: 13 Search results Filter:

Accession	Name	Description	Organism
A1YF08	PDX1_GORGO	Pancreas/duodenum homeobox protein 1 (Homeodo...	Gorilla gorilla gorilla (Lowland gorilla)
A1YG85	PDX1_PANPA	Pancreas/duodenum homeobox protein 1 (Homeodo...	Pan paniscus (Pygmy chimpanzee) (...
A2T756	PDX1_PANTR	Pancreas/duodenum homeobox protein 1 (Homeodo...	Pan troglodytes (Chimpanzee)
Q0IIL7	CPLX1_BOVIN	Complexin-1	Bos taurus (Bovine)
Q0VCW1	SPOP_BOVIN	Speckle-type POZ protein	Bos taurus (Bovine)
Q24K02	IDE_BOVIN	Insulin-degrading enzyme (EC 3	Bos taurus (Bovine)
Q32L79	INSL6_BOVIN	Insulin-like peptide INSL6 precursor (Insulin-like pep...	Bos taurus (Bovine)
Q3TDX8|Q3TJH3|Q3...	NB5R4_MOUSE	Cytochrome b5 reductase 4 (EC 1	Mus musculus (Mouse)
Q3UFB7	NTRK1_MOUSE	High affinity nerve growth factor receptor precurso...	Mus musculus (Mouse)
Q5NVK7|Q5R4U6|Q5...	SPOP_PONPY	Speckle-type POZ protein	Pongo pygmaeus (Orangutan)
Q63132|Q63130|Q6...	ROS_RAT	Proto-oncogene tyrosine-protein kinase ROS precur...	Rattus norvegicus (Rat)
Q78DX7|Q60705	ROS_MOUSE	Proto-oncogene tyrosine-protein kinase ROS precur...	Mus musculus (Mouse)
Q8TL28	ULIL_METAC	Ulilysin precursor (EC 3	Methanosarcina acetivorans

Total number of hits: 13

Figure 11.3: *The UniProt search view.*

11.2.1 UniProt search options

Conducting a search in **UniProt** from *CLC Genomics Workbench* corresponds to conducting the search on UniProt's website. When conducting the search from *CLC Genomics Workbench*, the results are available and ready to work with straight away.

Above the search fields, you can choose which database to search:

- **Swiss-Prot** This is believed to be the most accurate and best quality protein database available. All entries in the database has been currated manually and data are entered according to the original research paper.
- **TrEMBL** This database contain computer annotated protein sequences, thus the quality of the annotations is not as good as the Swiss-Prot database.

As default, *CLC Genomics Workbench* offers one text field where the search parameters can be entered. Click **Add search parameters** to add more parameters to your search.
Note! The search is a "and" search, meaning that when adding search parameters to your search, you search for both (or all) text strings rather than "any" of the text strings.

You can append a wildcard character by checking the checkbox at the bottom. This means that you only have to enter the first part of the search text, e.g. searching for "genom" will find both "genomic" and "genome".

The following parameters can be added to the search:

- **All fields**. Text, searches in all parameters in the UniProt database at the same time.
- **Organism**. Text.
- **Description**. Text.
- **Created Since**. Between 30 days and 10 years.
- **Feature**. Text.

The search parameters listed in the dialog are the most recently used. The **All fields** allows searches in all parameters in the UniProt database at the same time.

When you are satisfied with the parameters you have entered, click **Start search**.

Note! When conducting a search, no files are downloaded. Instead, the program produces a list of links to the files in the UniProt database. This ensures a much faster search.

11.2.2 Handling of UniProt search results

The search result is presented as a list of links to the files in the UniProt database. The **View** displays 50 hits at a time (can be changed in the **Preferences** (see chapter 5). More hits can be displayed by clicking the **More...** button at the bottom right of the **View**. More hits can be displayed by clicking the **More...** button at the bottom left of the **View**.

Each sequence hit is represented by text in three columns:

- Accession
- Name
- Description
- Organism
- Length.

It is possible to exclude one or more of these columns by adjust the View preferences for the database search view. Furthermore, your changes in the View preferences can be saved. See section 5.6.

Several sequences can be selected, and by clicking the buttons in the bottom of the search view, you can do the following:

- Download and open, does not save the sequence.
- Download and save, lets you choose location for saving sequence.
- Open at UniProt, searches the sequence at UniProt's web page.

Double-clicking a hit will download and open the sequence. The hits can also be copied into the **View Area** or the **Navigation Area** from the search results by drag and drop, copy/paste or by using the right-click menu as described below.

Drag and drop from UniProt search results

The sequences from the search results can be opened by dragging them into a position in the **View Area**.

Note! A sequence is not saved until the **View** displaying the sequence is closed. When that happens, a dialog opens: Save changes of sequence x? (Yes or No).

The sequence can also be saved by dragging it into the **Navigation Area**. It is possible to select more sequences and drag all of them into the **Navigation Area** at the same time.

Download UniProt search results using right-click menu

You may also select one or more sequences from the list and download using the right-click menu (see figure 11.2). Choosing **Download and Save** lets you select a folder or location where the sequences are saved when they are downloaded. Choosing **Download and Open** opens a new view for each of the selected sequences.

Copy/paste from UniProt search results

When using copy/paste to bring the search results into the **Navigation Area**, the actual files are downloaded from UniProt.

To copy/paste files into the **Navigation Area**:

> **select one or more of the search results | Ctrl + C (⌘ + C on Mac) | select location or folder in the Navigation Area | Ctrl + V**

Note! Search results are downloaded before they are saved. Downloading and saving several files may take some time. However, since the process runs in the background (displayed in the **Toolbox** under the **Processes** tab) it is possible to continue other tasks in the program. Like the search process, the download process can be stopped, paused, and resumed.

11.2.3 Save UniProt search parameters

The search view can be saved either using dragging the search tab and and dropping it in the **Navigation Area** or by clicking **Save** (💾). When saving the search, only the parameters are saved - not the results of the search. This is useful if you have a special search that you perform from time to time.

Even if you don't save the search, the next time you open the search view, it will remember the parameters from the last time you did a search.

11.3 Search for structures at NCBI

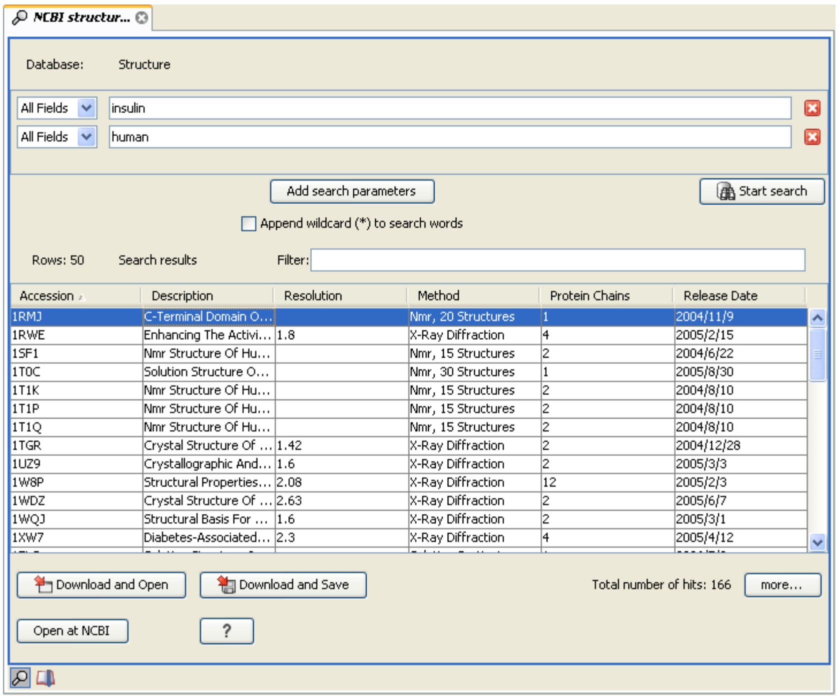
This section describes searches for three dimensional structures from the NCBI structure database http://www.ncbi.nlm.nih.gov/Structure/MMDB/mmdb.shtml. For manipulating and visualization of the downloaded structures see section 13.

The NCBI search view is opened in this way:

> **Search | Search for structures at NCBI ()**

or **Ctrl + B (⌘ + B on Mac)**

This opens the view shown in figure 11.4:



Database: Structure

All Fields: insulin

All Fields: human

□ Append wildcard (*) to search words

Rows: 50 Search results Filter:

Accession	Description	Resolution	Method	Protein Chains	Release Date
1RMJ	C-Terminal Domain O...		Nmr, 20 Structures	1	2004/11/9
1RWE	Enhancing The Activi...	1.8	X-Ray Diffraction	4	2005/2/15
1SF1	Nmr Structure Of Hu...		Nmr, 15 Structures	2	2004/6/22
1T0C	Solution Structure O...		Nmr, 30 Structures	1	2005/8/30
1T1K	Nmr Structure Of Hu...		Nmr, 15 Structures	2	2004/8/10
1T1P	Nmr Structure Of Hu...		Nmr, 15 Structures	2	2004/8/10
1T1Q	Nmr Structure Of Hu...		Nmr, 15 Structures	2	2004/8/10
1TGR	Crystal Structure Of ...	1.42	X-Ray Diffraction	2	2004/12/28
1UZ9	Crystallographic And...	1.6	X-Ray Diffraction	2	2005/3/3
1W8P	Structural Properties...	2.08	X-Ray Diffraction	12	2005/2/3
1WDZ	Crystal Structure Of ...	2.63	X-Ray Diffraction	2	2005/6/7
1WQJ	Structural Basis For ...	1.6	X-Ray Diffraction	2	2005/3/1
1XW7	Diabetes-Associated...	2.3	X-Ray Diffraction	4	2005/4/12

Total number of hits: 166

Figure 11.4: *The structure search view.*

11.3.1 Structure search options

Conducting a search in the **NCBI Database** from *CLC Genomics Workbench* corresponds to conducting search for structures on the NCBI's Entrez website. When conducting the search from *CLC Genomics Workbench*, the results are available and ready to work with straight away.

As default, *CLC Genomics Workbench* offers one text field where the search parameters can be entered. Click **Add search parameters** to add more parameters to your search.

Note! The search is a "AND" search, meaning that when adding search parameters to your search, you search for both (or all) text strings rather than "any" of the text strings.

You can append a wildcard character by clicking the checkbox at the bottom. This means that you only have to enter the first part of the search text, e.g. searching for "prot" will find both "protein" and "protease".

The following parameters can be added to the search:

- **All fields**. Text, searches in all parameters in the NCBI structure database at the same time.
- **Organism**. Text.
- **Author**. Text.
- **PdbAcc**. The accession number of the structure in the PDB database.

The search parameters are the most recently used. The **All fields** allows searches in all parameters in the database at the same time.

All fields also provide an opportunity to restrict a search to parameters which are not listed in the dialog. E.g. writing 'gene[Feature key] AND mouse' in **All fields** generates hits in the GenBank database which contains one or more genes and where 'mouse' appears somewhere in GenBank file. NB: the 'Feature Key' option is only available in GenBank when searching for nucleotide structures. For more information about how to use this syntax, see http://www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary_Matrices.html#Search_Fields_and_Qualifiers

When you are satisfied with the parameters you have entered click **Start search**.

Note! When conducting a search, no files are downloaded. Instead, the program produces a list of links to the files in the NCBI database. This ensures a much faster search.

11.3.2 Handling of NCBI structure search results

The search result is presented as a list of links to the files in the NCBI database. The **View** displays 50 hits at a time (can be changed in the **Preferences** (see chapter 5). More hits can be displayed by clicking the **More...** button at the bottom right of the **View**.

Each structure hit is represented by text in three columns:

- Accession.
- Description.
- Resolution.
- Method.
- Protein chains
- Release date.

It is possible to exclude one or more of these columns by adjust the View preferences for the database search view. Furthermore, your changes in the View preferences can be saved. See section 5.6.

Several structures can be selected, and by clicking the buttons in the bottom of the search view, you can do the following:

- **Download and open.** Download and open immediately.
- **Download and save.** Download and save lets you choose location for saving structure.
- **Open at NCBI.** Open additional information on the selected structure at NCBI's web page.

Double-clicking a hit will download and open the structure. The hits can also be copied into the **View Area** or the **Navigation Area** from the search results by drag and drop, copy/paste or by using the right-click menu as described below.

Drag and drop from structure search results

The structures from the search results can be opened by dragging them into a position in the **View Area**.

Note! A structure is not saved until the **View** displaying the structure is closed. When that happens, a dialog opens: Save changes of structure x? (Yes or No).

The structure can also be saved by dragging it into the **Navigation Area**. It is possible to select more structures and drag all of them into the **Navigation Area** at the same time.

Download structure search results using right-click menu

You may also select one or more structures from the list and download using the right-click menu (see figure 11.5). Choosing **Download and Save** lets you select a folder or location where the structures are saved when they are downloaded. Choosing **Download and Open** opens a new view for each of the selected structures.

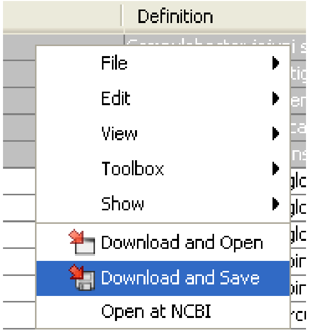


Figure 11.5: *By right-clicking a search result, it is possible to choose how to handle the relevant structure.*

The selected structures are not downloaded from the NCBI website but is downloaded from the RCSB Protein Data Bank `http://www.rcsb.org/pdb/home/home.do` in mmCIF format.

Copy/paste from structure search results

When using copy/paste to bring the search results into the **Navigation Area**, the actual files are downloaded.

To copy/paste files into the **Navigation Area**:

> **select one or more of the search results | Ctrl + C (⌘ + C on Mac) | select location or folder in the Navigation Area | Ctrl + V**

Note! Search results are downloaded before they are saved. Downloading and saving several files may take some time. However, since the process runs in the background (displayed in the **Status bar**) it is possible to continue other tasks in the program. Like the search process, the download process can be stopped. This is done in the **Toolbox** in the **Processes** tab.

11.3.3 Save structure search parameters

The search view can be saved either using dragging the search tab and and dropping it in the **Navigation Area** or by clicking **Save** (🖫). When saving the search, only the parameters are saved

- not the results of the search. This is useful if you have a special search that you perform from time to time.

Even if you don't save the search, the next time you open the search view, it will remember the parameters from the last time you did a search.

11.4 Sequence web info

CLC Genomics Workbench provides direct access to web-based search in various databases and on the Internet using your computer's default browser. You can look up a sequence in the databases of NCBI and UniProt, search for a sequence on the Internet using Google and search for Pubmed references at NCBI. This is useful for quickly obtaining updated and additional information about a sequence.

The functionality of these search functions depends on the information that the sequence contains. You can see this information by viewing the sequence as text (see section 10.5). In the following sections, we will explain this in further detail.

The procedure for searching is identical for all four search options (see also figure 11.6):

> **Open a sequence or a sequence list | Right-click the name of the sequence | Web Info (🌐) | select the desired search function**

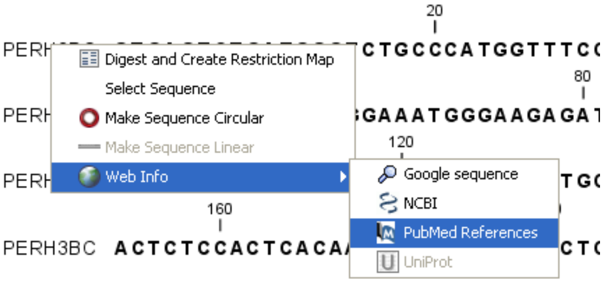


Figure 11.6: *Open webpages with information about this sequence.*

This will open your computer's default browser searching for the sequence that you selected.

11.4.1 Google sequence

The Google search function uses the accession number of the sequence which is used as search term on `http://www.google.com`. The resulting web page is equivalent to typing the accession number of the sequence into the search field on `http://www.google.com`.

11.4.2 NCBI

The NCBI search function searches in GenBank at NCBI (`http://www.ncbi.nlm.nih.gov`) using an identification number (when you view the sequence as text it is the "GI" number). Therefore, the sequence file must contain this number in order to look it up at NCBI. All sequences downloaded from NCBI have this number.

11.4.3 PubMed References

The PubMed references search option lets you look up Pubmed articles based on references contained in the sequence file (when you view the sequence as text it contains a number of "PUBMED" lines). Not all sequence have these PubMed references, but in this case you will se a dialog and the browser will not open.

11.4.4 UniProt

The UniProt search function searches in the UniProt database (http://www.ebi.uniprot.org) using the accession number. Furthermore, it checks whether the sequence was indeed downloaded from UniProt.

11.4.5 Additional annotation information

When sequences are downloaded from GenBank they often link to additional information on taxonomy, conserved domains etc. If such information is available for a sequence it is possible to access additional accurate online information. If the db_xref identifier line is found as part of the annotation information in the downloaded GenBank file, it is possible to easily look up additional information on the NCBI web-site.

To access this feature, simply right click an annotation and see which databases are available.

Chapter 12

BLAST search

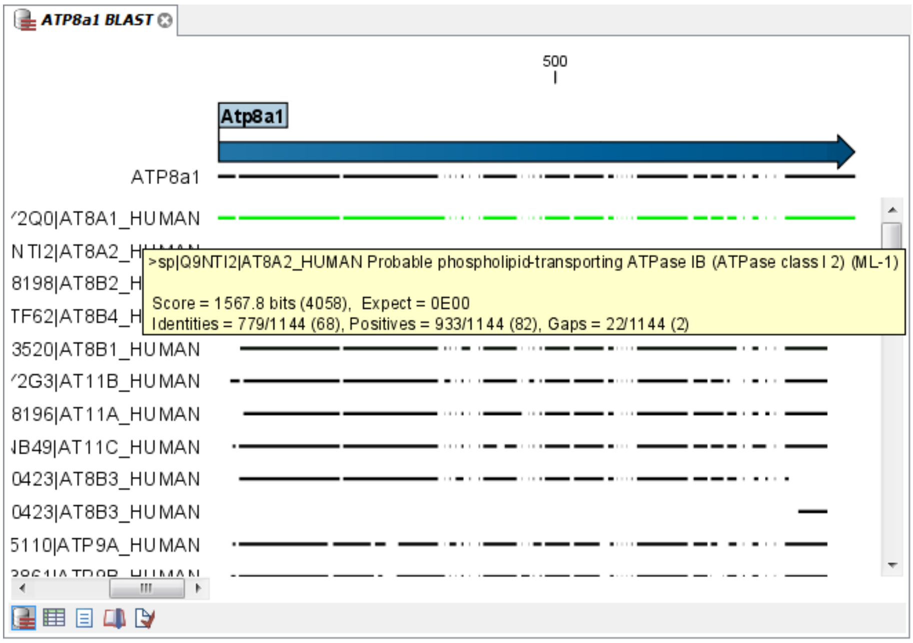
Contents

CLC Genomics Workbench offers to conduct BLAST searches on protein and DNA sequences. In short, a BLAST search identifies homologous sequences between your input (query) query sequence and a database of sequences [McGinnis and Madden, 2004]. BLAST (Basic Local Alignment Search Tool), identifies homologous sequences using a heuristic method which finds short matches between two sequences. After initial match BLAST attempts to start local alignments from these initial matches.

If you are interested in the bioinformatics behind BLAST, there is an easy-to-read explanation of this in section 12.6.

With *CLC Genomics Workbench* there are two ways of performing BLAST searches: You can either have the BLAST process run on NCBI's BLAST servers (`http://www.ncbi.nlm.nih.gov/`) or perform the BLAST search on your own computer. The advantage of running the BLAST search on NCBI servers is that you have readily access to the most popular BLAST databases without having to download them to your own computer. The advantage of running BLAST on your own computer is that you can use your own sequence data, and that this can sometimes be faster and more reliable for big batch BLAST jobs

Figure 12.1 shows an example of a BLAST result in the *CLC Genomics Workbench*.



>sp|Q9NTI2|AT8A2_HUMAN Probable phospholipid-transporting ATPase IB (ATPase class I 2) (ML-1)

Score = 1567.8 bits (4058), Expect = 0E00
Identities = 779/1144 (68), Positives = 933/1144 (82), Gaps = 22/1144 (2)

Figure 12.1: *Display of the output of a BLAST search. At the top is there a graphical representation of BLAST hits with tool-tips showing additional information on individual hits. Below is a tabular form of the BLAST results.*

12.1 BLAST at NCBI

When running BLAST at NCBI, the Workbench sends the sequences you select as input to NCBI's BLAST servers, and when the result is ready, it will be automatically downloaded and displayed in the Workbench. When you BLAST many sequences, the Workbench automatically splits the sequences up in smaller subsets and sends one subset at the time to NCBI. The size of the subset depends both on the number and size of the sequences.

To start the BLAST at NCBI:

Toolbox | **BLAST** ( ) | **NCBI BLAST** ( )

Alternatively, use the keyboard shortcut: Ctrl+Shift+B for Windows and ⌘ +Shift+B on Mac OS.

This opens the dialog seen in figure 12.2

Select one or more sequences of the same type (either DNA or protein) and click **Next**.

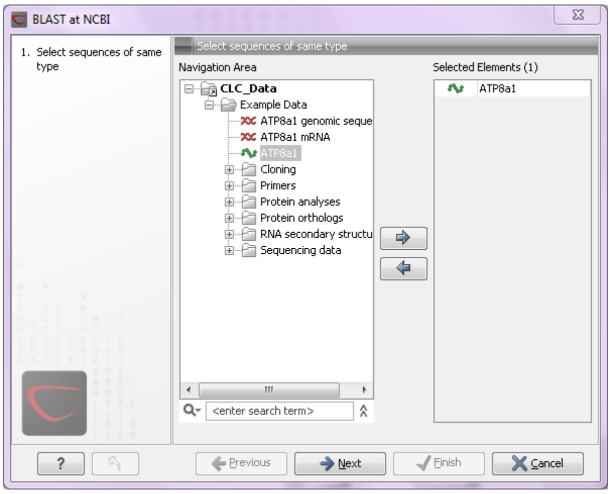
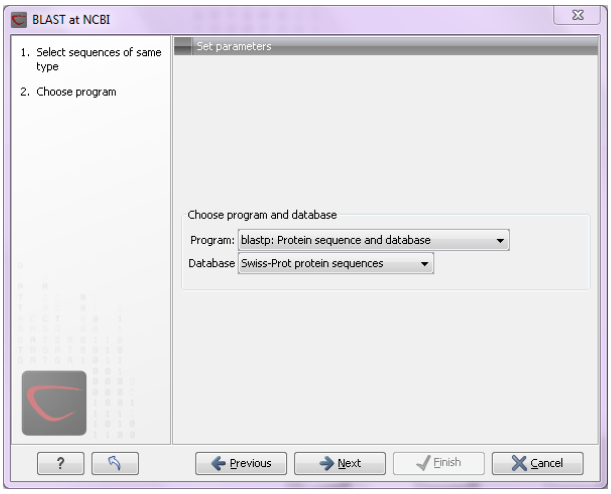


Figure 12.2: *Choose one or more sequences to conduct a BLAST search.*

In this dialog, you can choose which type of BLAST search you want to conduct, and you can limit your search to a particular database (see section D in the appendix for a list of available databases) as shown in figure 12.3:



Program: blastp: Protein sequence and database

Database Swiss-Prot protein sequences

Figure 12.3: *Choose a BLAST Program and a database for the search.*

BLAST search for DNA sequences:

- **BLASTn: DNA sequence against DNA database.** This BLAST method is used to identify homologous DNA sequences to your query sequence.
- **BLASTx: Translated DNA sequence against Protein database.** If you want to search in protein databases, this BLAST method allows for automated translation of the DNA input sequence and searching in various protein databases.
- **tBLASTx: Translated DNA sequence against Translated DNA database.** Here is both the input DNA sequence and the searched DNA database automatically translated.

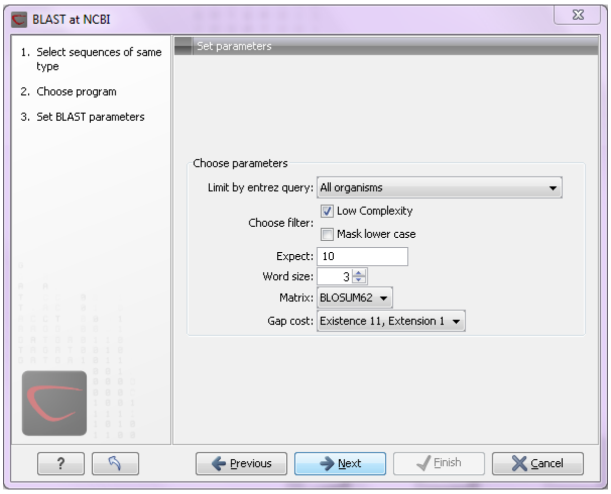
BLAST search for protein sequences:

- **BLASTp: Protein sequence against Protein database.** This the most common BLAST method used when searching for homologous protein sequences having a protein sequence as search input.
- **tBLASTn: Protein sequence against Translated DNA database.** Here is the protein sequence searched against an automatically translated DNA database.

Depending on whether you choose a protein or a DNA sequence, a number of different databases can be searched. A complete list of these databases can be found in Appendix D where you can also read how to add additional databases from NCBI.

If you search against the **Protein Data Bank protein** database homologous sequences are found to the query sequence, these can be downloaded and opened with the 3D view.

Click **Next** where you can limit the BLAST search by adjusting the parameters seen in figure 12.4.



1. Select sequences of same type
2. Choose program
3. Set BLAST parameters

Set parameters

Choose parameters

Limit by entrez query: All organisms

Choose filter: ☑ Low Complexity ☐ Mask lower case

Expect: 10

Word size: 3

Matrix: BLOSUM62

Gap cost: Existence 11, Extension 1

Figure 12.4: *Examples of different limitations which can be set before submitting a BLAST search.*

When choosing BLASTx or tBLASTx to conduct a search, you get the option of selecting a translation table for the genetic code. The standard genetic code is set as default. This is particularly useful when working with organisms or organelles which have a genetic code that differs from the standard genetic code.

The following description of BLAST search parameters is based on information from `http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml`.

- **Limit by Entrez query** BLAST searches can be limited to the results of an Entrez query against the database chosen. This can be used to limit searches to subsets of the BLAST databases. Any terms can be entered that would normally be allowed in an Entrez search session. Some queries are pre-entered and can be chosen in the drop down menu.
- **Choose filter**
  - **Low-complexity.** Mask off segments of the query sequence that have low compositional complexity. Filtering can eliminate statistically significant, but biologically uninteresting reports from the BLAST output (e.g. hits against common acidic-, basic- or proline-rich regions), leaving the more biologically interesting regions of the query sequence available for specific matching against database sequences.

  - **Mask lower case.** With this option selected you can cut and paste a FASTA sequence in upper case characters and denote areas you would like filtered with lower case. This allows you to customize what is filtered from the sequence during the comparison to the BLAST databases

- **Expect.** The statistical significance threshold for reporting matches against database sequences: the default value is 10, meaning that 10 matches are expected to be found merely by chance, according to the stochastic model of Karlin and Altschul (1990). If the statistical significance ascribed to a match is greater than the EXPECT threshold, the match will not be reported. Lower EXPECT thresholds are more stringent, leading to fewer chance matches being reported. Increasing the threshold shows less stringent matches. Fractional values are acceptable.

- **Word Size.** BLAST is a heuristic that works by finding word-matches between the query and database sequences. You may think of this process as finding "hot-spots" that BLAST can then use to initiate extensions that might lead to full-blown alignments. For nucleotide-nucleotide searches (i.e. "BLASTn") an exact match of the entire word is required before an extension is initiated, so that you normally regulate the sensitivity and speed of the search by increasing or decreasing the wordsize. For other BLAST searches non-exact word matches are taken into account based upon the similarity between words. The amount of similarity can be varied so that you normally uses just the wordsizes 2 and 3 for these searches.

- **Matrix.** A key element in evaluating the quality of a pairwise sequence alignment is the "substitution matrix", which assigns a score for aligning any possible pair of residues. The matrix used in a BLAST search can be changed depending on the type of sequences you are searching with (see the BLAST Frequently Asked Questions). Only applicable for protein sequences or translated DNA sequences.

- **Gap Cost.** The pull down menu shows the Gap Costs (Penalty to open Gap and penalty to extend Gap). Increasing the Gap Costs and Lambda ratio will result in alignments which decrease the number of Gaps introduced.

The more limitations are submitted to the search parameters, the faster the search will be conducted. If no limitations are submitted, the BLAST search may take several minutes.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

12.1.1 BLAST a selection against NCBI

If you only wish to BLAST a part of a sequence, this is possible directly from the sequence view:

> **select the region that you wish to BLAST | right-click the selection | BLAST Selection Against NCBI ( )**

This will go directly to the dialog shown in figure 12.3 and the rest of the options are the same as when performing a BLAST search with a full sequence.

12.2 BLAST against local data

Running BLAST searches locally is an advantage in the following situations:

- It will often be faster and more reliable than using NCBI BLAST
- You can use longer sequences
- You use your own data sets as database for BLASTing

On a technical level, the *CLC Genomics Workbench* uses the BLAST programs available from NCBI (see ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/), so the results of running the same data set locally and at NCBI will be the same.

As database, you can either create a database based on data in the **Navigation Area** (see section 12.4), you can download pre-formatted databases (see section 12.5.1), or you can use sequence data from the **Navigation Area** directly without creating a database first.

To conduct a BLAST search:

or **Toolbox** | **BLAST** ( ) | **Local BLAST** ( )

This opens the dialog seen in figure 12.5:

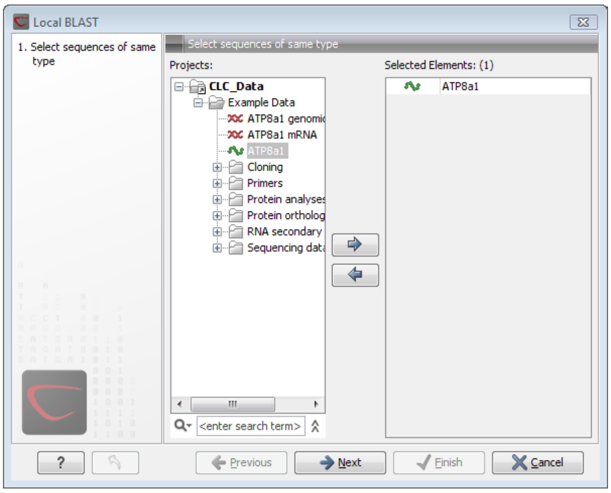


Figure 12.5: *Choose one or more sequences to conduct a BLAST search.*

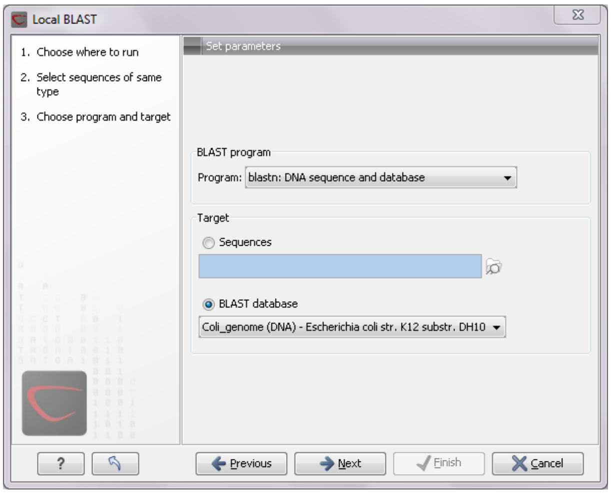
Select one or more sequences of the same type (DNA or protein) and click **Next**.

This opens the dialog seen in figure 12.6:

At the top, you can choose between different BLAST programs. See section 12.1 for information about these methods.

Below, you specify the target database to use:

- **Sequences**. When you choose this option, you can add sequence data from the **Navigation Area** as database by clicking the **Browse and select** icon ( ). This will automatically create a temporary BLAST database from these sequences that will be used for the BLAST search and deleted afterwards. If you want to be able to click in the BLAST result to retrieve the hit sequences from the BLAST database at a later point, you should not use this option; create a create a BLAST database first, see section 12.4.
- **BLAST Database**. Select a database that has already been created. Read more in section 12.5.



BLAST program

Program: blastn: DNA sequence and database

Target

○ Sequences

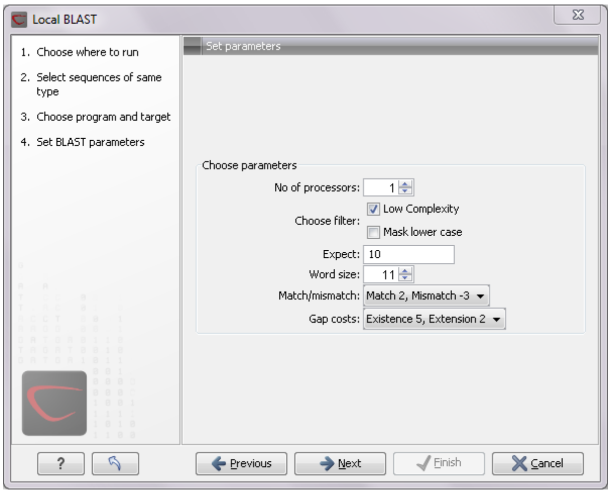
● BLAST database

Coli_genome (DNA) - Escherichia coli str. K12 substr. DH10

Figure 12.6: *Choose a BLAST program and a target database to conduct BLAST search.*

When a database or a set of sequences has been selected, click **Next**.

This opens the dialog seen in figure 12.7:



Choose parameters

No of processors: 1

Choose filter: ☑ Low Complexity ☐ Mask lower case

Expect: 10

Word size: 11

Match/mismatch: Match 2, Mismatch -3

Gap costs: Existence 5, Extension 2

Figure 12.7: *Examples of different limitations which can be set before submitting a BLAST search.*

See section 12.1 for information about these limitations.

There is one additional setting that is not available for BLAST at NCBI:

- **Number of processors.** It is possible to specify the number of processors which should be used if the Workbench is installed on a multi-processor system.

12.2.1 BLAST a selection against a local database

If you only wish to BLAST a part of a sequence, this is possible directly from the sequence view:

> **select the region that you wish to BLAST | right-click the selection | BLAST Selection Against Local Database ()**

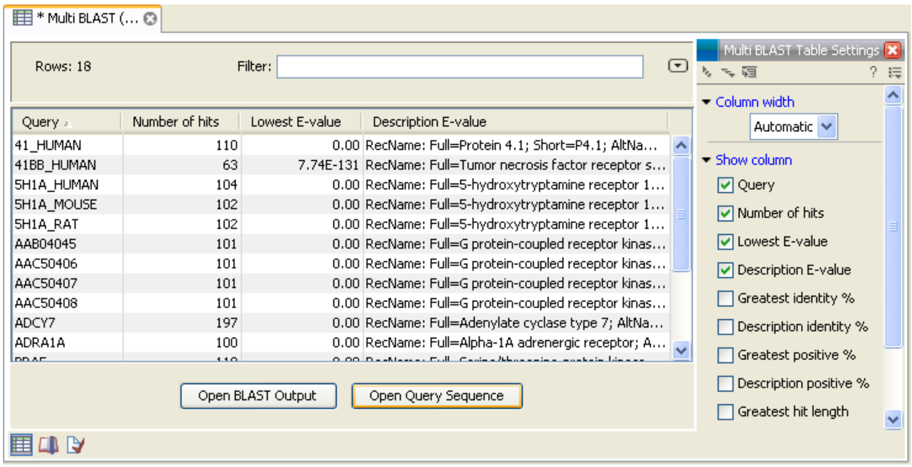
This will go directly to the dialog shown in figure 12.6 and the rest of the options are the same as when performing a BLAST search with a full sequence.

12.3 Output from BLAST search

The output of the BLAST search is similar for both BLAST at NCBI and the local BLAST. If more than one sequence was selected as input, the result will be an overview BLAST table, summarizing the results for each input sequence and allowing access to the detailed result of each sequence. Both the overview BLAST table and the detailed BLAST results are described below.

12.3.1 Overview BLAST table

In the overview BLAST table shown in figure 12.8, there is one row for each query sequence. Each row represents the BLAST result for this query sequence.



Rows: 18

Filter:

Query	Number of hits	Lowest E-value	Description E-value
41_HUMAN	110	0.00	RecName: Full=Protein 4.1; Short=P4.1; AltNa...
41BB_HUMAN	63	7.74E-131	RecName: Full=Tumor necrosis factor receptor s...
5H1A_HUMAN	104	0.00	RecName: Full=5-hydroxytryptamine receptor 1...
5H1A_MOUSE	102	0.00	RecName: Full=5-hydroxytryptamine receptor 1...
5H1A_RAT	102	0.00	RecName: Full=5-hydroxytryptamine receptor 1...
AAB04045	101	0.00	RecName: Full=G protein-coupled receptor kinas...
AAC50406	101	0.00	RecName: Full=G protein-coupled receptor kinas...
AAC50407	101	0.00	RecName: Full=G protein-coupled receptor kinas...
AAC50408	101	0.00	RecName: Full=G protein-coupled receptor kinas...
ADCY7	197	0.00	RecName: Full=Adenylate cyclase type 7; AltNa...
ADRA1A	100	0.00	RecName: Full=Alpha-1A adrenergic receptor; A...

Column width: Automatic

☑ Query
☑ Number of hits
☑ Lowest E-value
☑ Description E-value
☐ Greatest identity %
☐ Description identity %
☐ Greatest positive %
☐ Description positive %
☐ Greatest hit length

Figure 12.8: *An overview BLAST table summarizing the results for a number of query sequences.*

Double-clicking a row will open the BLAST result for this query sequence, allowing more detailed investigation of the result. You can also select one or more rows and click the **Open BLAST Output** button at the bottom of the view. Clicking the **Open Query Sequence** will open a sequence list with the selected query sequences. This can be useful in work flows where BLAST is used as a filtering mechanism where you can filter the table to include e.g. sequences that have a certain top hit and then extract those.

In the overview table, the following information is shown:

- Query: Since this table displays information about several query sequences, the first column is the name of the query sequence.
- Number of hits: The number of hits for this query sequence.
- Top hit E-value: The E-value of the top hit is shown here. The top hit is defined as the hit with the lowest E-value.
- Top hit: The description of the top hit. If there is no description, it will just be the name of the top hit.

If you wish to save some of the BLAST results as individual elements in the **Navigation Area**, open them and click **Save As** in the **File** menu.

12.3.2 BLAST graphics

The **BLAST editor** shows the sequences hits which were found in the BLAST search. The hit sequences are represented by colored horizontal lines, and when hovering the mouse pointer over a BLAST hit sequence, a tooltip appears, listing the characteristics of the sequence. As default, the query sequence is fitted to the window width, but it is possible to zoom in the windows and see the actual sequence alignments returned from the BLAST server.

There are several settings available in the **BLAST Graphics** view.

- **BLAST Layout.** You can choose to **Gather sequences at top**. Enabling this option affects the view that is shown when scrolling horizontally along a BLAST result. If selected, the sequence hits which did not contribute to the visible part of the BLAST graphics will be omitted whereas the found BLAST hits will automatically be placed right below the query sequence.
- **BLAST hit coloring.** You can choose whether to color hit sequences and you can adjust the coloring.
- **Compactness**: In the **Sequence Layout** in the Side Panel, you can control the level of sequence detail to be displayed:
  - **Not compact.** Full detail and spaces between the sequences.
  - **Low.** The normal settings where the residues are visible (when zoomed in) but with no extra spaces between.
  - **Medium.** The sequences are represented as lines and the residues are not visible. There is some space between the sequences.
  - **Compact.** Even less space between the sequences.
- **Coverage**: In the Alignment info in the Side Panel, you can visualize the number of hit sequences at a given position on the query sequence. The level of coverage is relative to the overall number of hits included in the result.
  - **Foreground color.** Colors the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage.
  - **Background color.** Colors the background of the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage
  - **Graph.** The coverage is displayed as a graph beneath the query sequence (Learn how to export the data behind the graph in section 7.4).
    - **Height.** Specifies the height of the graph.
    - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
    - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. If a Color bar is chosen, the color box is replaced by a gradient color box as described under Foreground color.

The remaining View preferences for BLAST Graphics are the same as those of alignments. See section 22.2.

Some of the information available in the tooltips is:

- **Name of sequence.** Here is shown some additional information of the sequence which was found. This line corresponds to the description line in GenBank (if the search was conducted on the nr database).
- **Score.** This shows the bit score of the local alignment generated through the BLAST search.
- **Expect.** Also known as the E-value. A low value indicates a homologous sequence. Higher E-values indicate that BLAST found a less homologous sequence.
- **Identities.** This number shows the number of identical residues or nucleotides in the obtained alignment.
- **Gaps.** This number shows whether the alignment has gaps or not.
- **Strand.** This is only valid for nucleotide sequences and show the direction of the aligned strands. Minus indicate a complementary strand.
- **Query.** This is the sequence (or part of the sequence) which you have used for the BLAST search.
- **Sbjct (subject).** This is the sequence found in the database.

The numbers of the query and subject sequences refer to the sequence positions in the submitted and found sequences. If the subject sequence has number 59 in front of the sequence, this means that 58 residues are found upstream of this position, but these are not included in the alignment.

By right clicking the sequence name in the Graphical BLAST output it is possible to download the full hits sequence from NCBI with accompanying annotations and information. It is also possible to just open the actual hit sequence in a new view.

12.3.3 BLAST table

In addition to the graphical display of a BLAST result, it is possible to view the BLAST results in a tabular view. In the tabular view, one can get a quick and fast overview of the results. Here you can also select multiple sequences and download or open all of these in one single step. Moreover, there is a link from each sequence to the sequence at NCBI. These possibilities are either available through a right-click with the mouse or by using the buttons below the table.

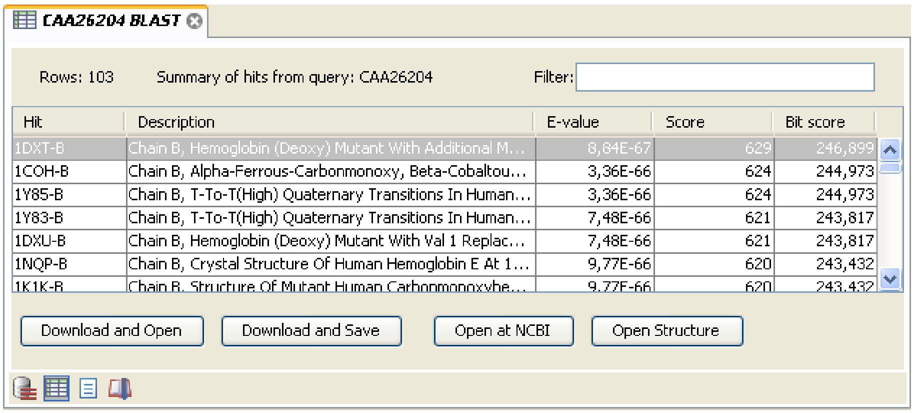
If the **BLAST table** view was not selected in **Step 4** of the BLAST search, the table can be shown in the following way:

Click the Show BLAST Table button (▦) at the bottom of the view

Figure 12.9 is an example of a BLAST Table.

The BLAST Table includes the following information:

- **Query sequence.** The sequence which was used for the search.
- **Hit.** The Name of the sequences found in the BLAST search.
- **Id.** GenBank ID.



Rows: 103 Summary of hits from query: CAA26204 Filter:

Hit	Description	E-value	Score	Bit score
1DXT-B	Chain B, Hemoglobin (Deoxy) Mutant With Additional M...	8,84E-67	629	246,899
1COH-B	Chain B, Alpha-Ferrous-Carbonmonoxy, Beta-Cobaltou...	3,36E-66	624	244,973
1Y85-B	Chain B, T-To-T(High) Quaternary Transitions In Human...	3,36E-66	624	244,973
1Y83-B	Chain B, T-To-T(High) Quaternary Transitions In Human...	7,48E-66	621	243,817
1DXU-B	Chain B, Hemoglobin (Deoxy) Mutant With Val 1 Replac...	7,48E-66	621	243,817
1NQP-B	Chain B, Crystal Structure Of Human Hemoglobin E At 1...	9,77E-66	620	243,432
1K1K-B	Chain B, Structure Of Mutant Human Carbonmonoxyhe...	9,77E-66	620	243,432

Figure 12.9: *Display of the output of a BLAST search in the tabular view. The hits can be sorted by the different columns, simply by clicking the column heading.*

- **Description.** Text from NCBI describing the sequence.
- **E-value.** Measure of quality of the match. Higher E-values indicate that BLAST found a less homologous sequence.
- **Score.** This shows the score of the local alignment generated through the BLAST search.
- **Bit score.** This shows the bit score of the local alignment generated through the BLAST search. Bit scores are normalized, which means that the bit scores from different alignments can be compared, even if different scoring matrices have been used.
- **Hit start.** Shows the start position in the hit sequence
- **Hit end.** Shows the end position in the hit sequence.
- **Hit length.** The length of the hit.
- **Query start.** Shows the start position in the query sequence.
- **Query end.** Shows the end position in the query sequence.
- **Overlap.** Display a percentage value for the overlap of the query sequence and hit sequence. Only the length of the local alignment is taken into account and not the full length query sequence.
- **Identity.** Shows the number of identical residues in the query and hit sequence.
- **%Identity.** Shows the percentage of identical residues in the query and hit sequence.
- **Positive.** Shows the number of similar but not necessarily identical residues in the query and hit sequence.
- **%Positive.** Shows the percentage of similar but not necessarily identical residues in the query and hit sequence.
- **Gaps.** Shows the number of gaps in the query and hit sequence.
- **%Gaps.** Shows the percentage of gaps in the query and hit sequence.
- **Query Frame/Strand.** Shows the frame or strand of the query sequence.

- **Hit Frame/Strand.** Shows the frame or strand of the hit sequence.

In the **BLAST table** view you can handle the hit sequences. Select one or more sequences from the table, and apply one of the following functions.

- **Download and Open.** Download the full sequence from NCBI and opens it. If multiple sequences are selected, they will all open (if the same sequence is listed several times, only one copy of the sequence is downloaded and opened).
- **Download and Save.** Download the full sequence from NCBI and save it. When you click the button, there will be a save dialog letting you specify a folder to save the sequences. If multiple sequences are selected, they will all open (if the same sequence is listed several times, only one copy of the sequence is downloaded and opened).
- **Open at NCBI.** Opens the corresponding sequence(s) at GenBank at NCBI. Here is stored additional information regarding the selected sequence(s). The default Internet browser is used for this purpose.
- **Open structure.** If the hit sequence contain structure information, the sequence is opened in a text view or a 3D view (3D view in CLC Protein Workbench and CLC Main Workbench).

You can do a text-based search in the information in the BLAST table by using the filter at the upper right part of the view. In this way you can search for e.g. species or other information which is typically included in the "Description" field.

The table is integrated with the graphical view described in section 12.3.2 so that selecting a hit in the table will make a selection on the corresponding sequence in the graphical view.

12.4 Create local BLAST database

In *CLC Genomics Workbench* you can create a local database which you can use for local BLAST. You can specify a location on your computer for saving the BLAST database files, and the Workbench will automatically list the BLAST database when running local BLAST (see section 12.2).

Both DNA, RNA, and protein sequences that are located in the **Navigation Area** can be used for the BLAST database. If you wish to use a pre-formatted BLAST database instead, see section 12.5.1.

To create the BLAST database:

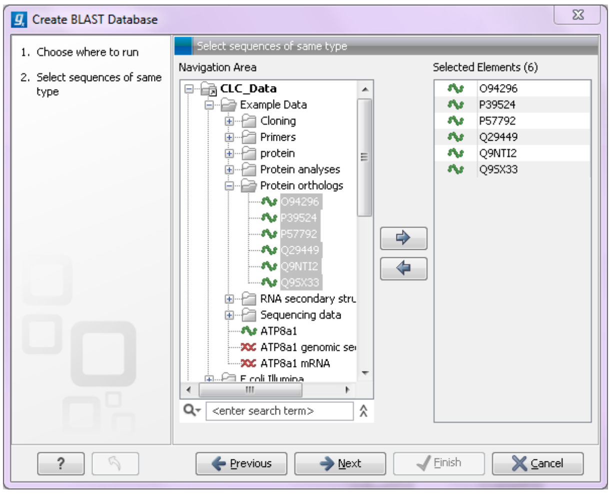
Toolbox | **BLAST ()** | **Create BLAST Database ()**

This opens the dialog seen in figure 12.10

Select sequences or sequence lists for the database an click **Next**.

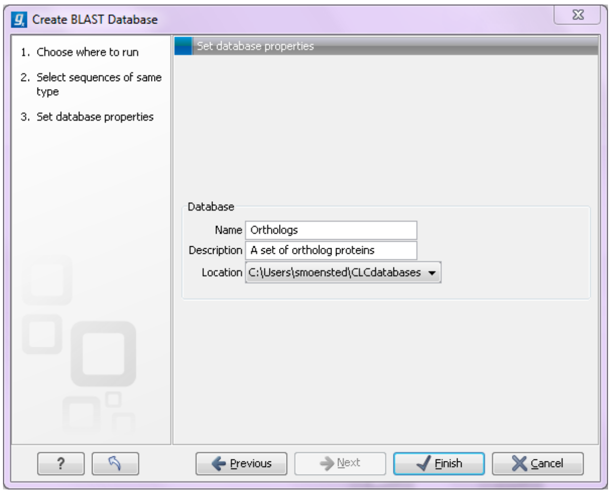
In the next dialog, shown in figure 12.11, you can provide more information about the database. The following information can be entered:

- **Name.** The name of the BLAST database. This name will be used when running BLAST and also as file name for the BLAST database files.



Selected Elements (6): O94296, P39524, P57792, Q29449, Q9NTI2, Q9SX33

Figure 12.10: *Add sequences for the BLAST database.*



Name: Orthologs

Description: A set of ortholog proteins

Location: C:\Users\smoensted\CLCdatabases

Figure 12.11: *Providing name, description and save location for the database.*

- **Description.** Here you can add more details to describe the contents of the database.
- **Location.** Select the location for saving the BLAST database files. You can add or change locations in the **Manage BLAST Databases** dialog, see section 12.5.

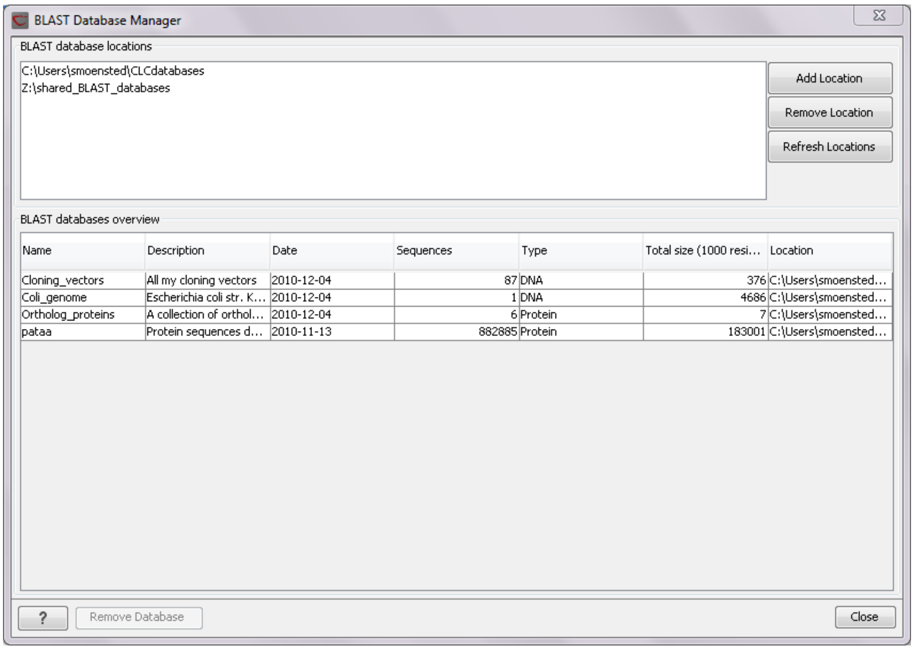
Click **Finish** to create the BLAST database. Once the process is complete, the new database will be available in the **Manage BLAST Databases** dialog, see section 12.5 and when running local BLAST (see section 12.2).

12.5 Manage BLAST databases

The BLAST database available as targets for running local BLAST (see section 12.2) can be managed through the Manage BLAST Databases dialog (see figure 12.12):

Toolbox | **BLAST** ( ) | **Manage BLAST Databases** ( )

At the top of the dialog, there is a list of the **BLAST database locations**. These locations are folders where the Workbench will look for valid BLAST databases. These can either be created from the Workbench using the **Create BLAST Database tool**, see section 12.4, or they can be



```
C:\Users\smoensted\CLCdatabases
Z:\shared_BLAST_databases
```

Name	Description	Date	Sequences	Type	Total size (1000 resi...	Location
Cloning_vectors	All my cloning vectors	2010-12-04	87	DNA	376	C:\Users\smoensted...
Coli_genome	Escherichia coli str. K...	2010-12-04	1	DNA	4686	C:\Users\smoensted...
Ortholog_proteins	A collection of orthol...	2010-12-04	6	Protein	7	C:\Users\smoensted...
pataa	Protein sequences d...	2010-11-13	882885	Protein	183001	C:\Users\smoensted...

Figure 12.12: *Overview of available BLAST databases.*

pre-formatted databases e.g. downloaded from NCBI (see below).

The list of locations can be modified using the **Add Location** and **Remove Location** buttons. Once the Workbench has scanned the locations, it will keep a cache of the databases (in order to improve performance). If you are in a situation where you have added new databases that are not listed, you can press **Refresh Locations** to clear the cache and search the locations again.

The default location is scanned recursively through all subfolders, but all other locations are only scanned at the top level.

Below the list of locations, all the BLAST databases are listed with the following information:

- **Name.** The name of the BLAST database.
- **Description.** Detailed description of the contents of the database.
- **Date.** The date the database was created.
- **Sequences.** The number of sequences in the database.
- **Type.** The type can be either nucleotide (DNA) or protein.
- **Total size (1000 residues).** The number of residues in the database, either bases or amino acid.
- **Location.** The location of the database.

Below the list of BLAST databases, there is a button to **Remove Database**. This option will delete the database files belonging to the database selected.

12.5.1 Add pre-formatted BLAST databases

Pre-formatted databases can be found for download many places, for example at ftp://ftp.ncbi.nih.gov/blast/db/. In order to use databases that have been downloaded or created outside the Workbench, either put the files in one of the locations defined in the BLAST database manager (see section 12.5), or add a new location for the folder where these database files are located.

Note that some databases depend on other databases to be complete. This is explained in the README file found in the link above.

12.5.2 Migrating from a previous version of the Workbench

In versions before *CLC Genomics Workbench 4.5*, the BLAST database management was very different from this. In order to migrate from the older versions, please add the folders of the old BLAST databases as locations in the BLAST database manager (see section 12.5). The old representations of the BLAST databases in the **Navigation Area** can be deleted.

If you have saved the BLAST databases in the default folder, they will automatically appear because the default database location used in *CLC Genomics Workbench 4.5* is the same as the default folder specified for saving BLAST databases in the old version.

12.6 Bioinformatics explained: BLAST

BLAST (Basic Local Alignment Search Tool) has become the *defacto* standard in search and alignment tools [Altschul et al., 1990]. The BLAST algorithm is still actively being developed and is one of the most cited papers ever written in this field of biology. Many researchers use BLAST as an initial screening of their sequence data from the laboratory and to get an idea of what they are working on. BLAST is far from being basic as the name indicates; it is a highly advanced algorithm which has become very popular due to availability, speed, and accuracy. In short, a BLAST search identifies homologous sequences by searching one or more databases usually hosted by NCBI (http://www.ncbi.nlm.nih.gov/), on the query sequence of interest [McGinnis and Madden, 2004].

BLAST is an open source program and anyone can download and change the program code. This has also given rise to a number of BLAST derivatives; WU-BLAST is probably the most commonly used [Altschul and Gish, 1996].

BLAST is highly scalable and comes in a number of different computer platform configurations which makes usage on both small desktop computers and large computer clusters possible.

12.6.1 Examples of BLAST usage

BLAST can be used for a lot of different purposes. A few of them are mentioned below.

- **Looking for species.** If you are sequencing DNA from unknown species, BLAST may help identify the correct species or homologous species.

- **Looking for domains.** If you BLAST a protein sequence (or a translated nucleotide sequence) BLAST will look for known domains in the query sequence.

- **Looking at phylogeny.** You can use the BLAST web pages to generate a phylogenetic tree of the BLAST result.
- **Mapping DNA to a known chromosome.** If you are sequencing a gene from a known species but have no idea of the chromosome location, BLAST can help you. BLAST will show you the position of the query sequence in relation to the hit sequences.
- **Annotations.** BLAST can also be used to map annotations from one organism to another or look for common genes in two related species.

12.6.2 Searching for homology

Most research projects involving sequencing of either DNA or protein have a requirement for obtaining biological information of the newly sequenced and maybe unknown sequence. If the researchers have no prior information of the sequence and biological content, valuable information can often be obtained using BLAST. The BLAST algorithm will search for homologous sequences in predefined and annotated databases of the users choice.

In an easy and fast way the researcher can gain knowledge of gene or protein function and find evolutionary relations between the newly sequenced DNA and well established data.

After the BLAST search the user will receive a report specifying found homologous sequences and their local alignments to the query sequence.

12.6.3 How does BLAST work?

BLAST identifies homologous sequences using a heuristic method which initially finds short matches between two sequences; thus, the method does not take the entire sequence space into account. After initial match, BLAST attempts to start local alignments from these initial matches. This also means that BLAST does not guarantee the optimal alignment, thus some sequence hits may be missed. In order to find optimal alignments, the Smith-Waterman algorithm should be used (see below). In the following, the BLAST algorithm is described in more detail.

Seeding

When finding a match between a query sequence and a hit sequence, the starting point is the *words* that the two sequences have in common. A word is simply defined as a number of letters. For blastp the default word size is 3 *W=3*. If a query sequence has a QWRTG, the searched words are QWR, WRT, RTG. See figure 12.13 for an illustration of words in a protein sequence.

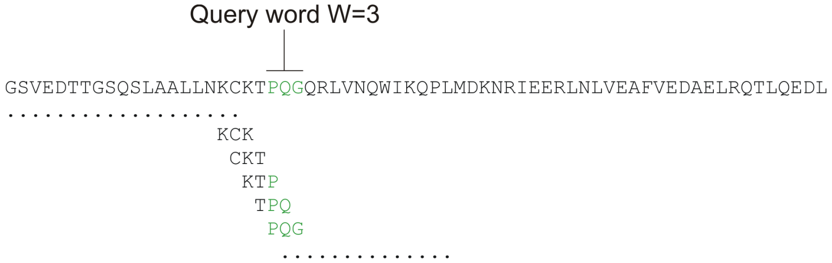

Query word W=3
GSVEDTTGSQSLAALLNKCKTPQGQRLVNQWIKQPLMDKNRIEERLNLVEAFVEDAELRQTLQEDL
....
KCK
CKT
KTP
TPQ
PQG
....

Figure 12.13: *Generation of exact BLAST words with a word size of W=3.*

During the initial BLAST seeding, the algorithm finds all common words between the query sequence and the hit sequence(s). Only regions with a word hit will be used to build on an alignment.

BLAST will start out by making words for the entire query sequence (see figure 12.13). For each word in the query sequence, a compilation of neighborhood words, which exceed the threshold of *T*, is also generated.

A neighborhood word is a word obtaining a score of at least *T* when comparing, using a selected scoring matrix (see figure 12.14). The default scoring matrix for blastp is BLOSUM62 (for explanation of scoring matrices, see www.clcbio.com/be). The compilation of exact words and neighborhood words is then used to match against the database sequences.

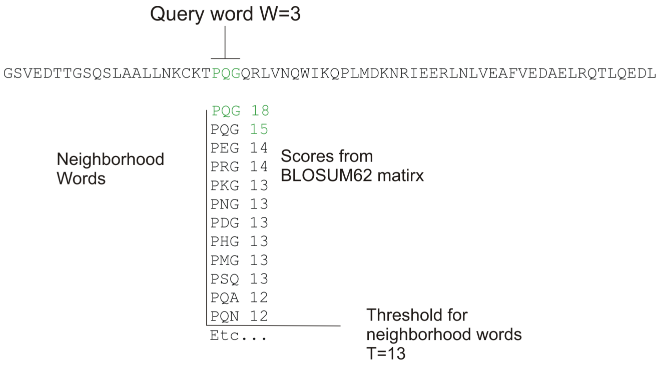

Query word W=3
GSVEDTTGSQSLAALLNKCKTPQGQRLVNQWIKQPLMDKNRIEERLNLVEAFVEDAELRQTLQEDL
Neighborhood Words
PQG 18
PQG 15
PEG 14
PRG 14
PKG 13
PNG 13
PDG 13
PHG 13
PMG 13
PSQ 13
PQA 12
PQN 12
Etc...
Scores from BLOSUM62 matirx
Threshold for neighborhood words T=13

Figure 12.14: *Neighborhood BLAST words based on the BLOSUM62 matrix. Only words where the threshold* T *exceeds 13 are included in the initial seeding.*

After initial finding of words (seeding), the BLAST algorithm will extend the (only 3 residues long) alignment in both directions (see figure 12.15). Each time the alignment is extended, an alignment score is increases/decreased. When the alignment score drops below a predefined threshold, the extension of the alignment stops. This ensures that the alignment is not extended to regions where only very poor alignment between the query and hit sequence is possible. If the obtained alignment receives a score above a certain threshold, it will be included in the final BLAST result.

```
Query: 325 SLAALLNKCKTPQGQRLVNQWIKQPLMDKNRIEERLNLVEA     365
           +LA++L+    TP G R++ +W+   P+ D    + ER    + A
Sbjct: 290 TLASVLDCTVTPMGSRMLKRWLHMPVRDTRVLLERQQTIGA     330
```

Figure 12.15: *Blast aligning in both directions. The initial word match is marked green.*

By tweaking the word size *W* and the neighborhood word threshold *T*, it is possible to limit the search space. E.g. by increasing *T*, the number of neighboring words will drop and thus limit the search space as shown in figure 12.16.

This will increase the speed of BLAST significantly but may result in loss of sensitivity. Increasing the word size *W* will also increase the speed but again with a loss of sensitivity.

12.6.4 Which BLAST program should I use?

Depending on the nature of the sequence it is possible to use different BLAST programs for the database search. There are five versions of the BLAST program, blastn, blastp, blastx, tblastn, tblastx:

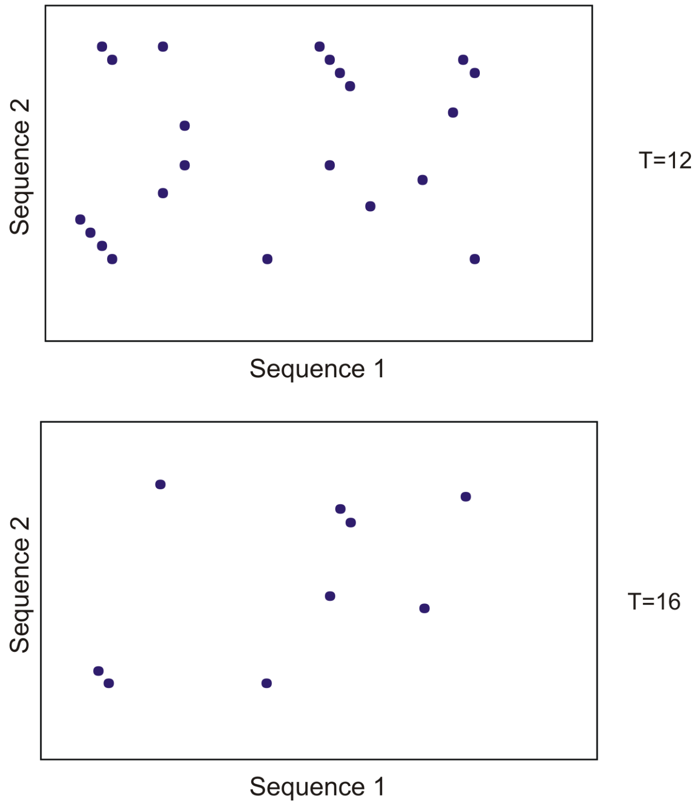


Figure 12.16: *Each dot represents a word match. Increasing the threshold of* T *limits the search space significantly.*

Option	**Query Type**	**DB Type**	**Comparison**	**Note**
blastn	Nucleotide	Nucleotide	Nucleotide-Nucleotide	
blastp	Protein	Protein	Protein-Protein	
tblastn	Protein	Nucleotide	Protein-Protein	The database is translated into protein
blastx	Nucleotide	Protein	Protein-Protein	The queries are translated into protein
tblastx	Nucleotide	Nucleotide	Protein-Protein	The queries and database are translated into protein

The most commonly used method is to BLAST a nucleotide sequence against a nucleotide database (blastn) or a protein sequence against a protein database (blastp). But often another BLAST program will produce more interesting hits. E.g. if a nucleotide sequence is translated before the search, it is more likely to find better and more accurate hits than just a blastn search. One of the reasons for this is that protein sequences are evolutionarily more conserved than nucleotide sequences. Another good reason for translating the query sequence before the search is that you get protein hits which are likely to be annotated. Thus you can directly see the protein function of the sequenced gene.

12.6.5 Which BLAST options should I change?

The NCBI BLAST web pages and the BLAST command line tool offer a number of different options which can be changed in order to obtain the best possible result. Changing these parameters can have a great impact on the search result. It is not the scope of this document to comment on all of the options available but merely the options which can be changed with a direct impact on the search result.

The E-value

The *expect value*(E-value) can be changed in order to limit the number of hits to the most significant ones. The lower the E-value, the better the hit. The E-value is dependent on the length of the query sequence and the size of the database. For example, an alignment obtaining an E-value of 0.05 means that there is a 5 in 100 chance of occurring by chance alone.

E-values are very dependent on the query sequence length and the database size. Short identical sequence may have a high E-value and may be regarded as "false positive" hits. This is often seen if one searches for short primer regions, small domain regions etc. The default threshold for the E-value on the BLAST web page is 10. Increasing this value will most likely generate more hits. Below are some rules of thumb which can be used as a guide but should be considered with common sense.

- **E-value < 10e-100** Identical sequences. You will get long alignments across the entire query and hit sequence.
- **10e-100 < E-value < 10e-50** Almost identical sequences. A long stretch of the query protein is matched to the database.
- **10e-50 < E-value < 10e-10** Closely related sequences, could be a domain match or similar.
- **10e-10 < E-value < 1** Could be a true homologue but it is a gray area.
- **E-value > 1** Proteins are most likely not related
- **E-value > 10** Hits are most likely junk unless the query sequence is very short.

Gap costs

For blastp it is possible to specify gap cost for the chosen substitution matrix. There is only a limited number of options for these parameters. The *open gap cost* is the price of introducing gaps in the alignment, and *extension gap cost* is the price of every extension past the initial opening gap. Increasing the gap costs will result in alignments with fewer gaps.

Filters

It is possible to set different filter options before running the BLAST search. Low-complexity regions have a very simple composition compared to the rest of the sequence and may result in problems during the BLAST search [Wootton and Federhen, 1993]. A low complexity region of a protein can for example look like this 'fftfflllsss', which in this case is a region as part of a signal peptide. In the output of the BLAST search, low-complexity regions will be marked in lowercase gray characters (default setting). The low complexity region cannot be thought of as a significant match; thus, disabling the low complexity filter is likely to generate more hits to sequences which are not truly related.

Word size

Change of the word size has a great impact on the seeded sequence space as described above. But one can change the word size to find sequence matches which would otherwise not be found using the default parameters. For instance the word size can be decreased when searching for primers or short nucleotides. For blastn a suitable setting would be to decrease the default word size of 11 to 7, increase the E-value significantly (1000) and turn off the complexity filtering.

For blastp a similar approach can be used. Decrease the word size to 2, increase the E-value and use a more stringent substitution matrix, e.g. a PAM30 matrix.

Fortunately, the optimal search options for finding short, nearly exact matches can already be found on the BLAST web pages http://www.ncbi.nlm.nih.gov/BLAST/.

Substitution matrix

For protein BLAST searches, a default substitution matrix is provided. If you are looking at distantly related proteins, you should either choose a high-numbered PAM matrix or a low-numbered BLOSUM matrix. See *Bioinformatics Explained* on scoring matrices on http://www.clcbio.com/be/. The default scoring matrix for blastp is BLOSUM62.

12.6.6 Explanation of the BLAST output

The BLAST output comes in different flavors. On the NCBI web page the default output is html, and the following description will use the html output as example. Ordinary text and xml output for easy computational parsing is also available.

The default layout of the NCBI BLAST result is a graphical representation of the hits found, a table of sequence identifiers of the hits together with scoring information, and alignments of the query sequence and the hits.

The graphical output (shown in figure 12.17) gives a quick overview of the query sequence and the resulting hit sequences. The hits are colored according to the obtained alignment scores.

The table view (shown in figure 12.18) provides more detailed information on each hit and furthermore acts as a hyperlink to the corresponding sequence in GenBank.

In the alignment view one can manually inspect the individual alignments generated by the BLAST algorithm. This is particularly useful for detailed inspection of the sequence hit found(sbjct) and the corresponding alignment. In the alignment view, all scores are described for each alignment, and the start and stop positions for the query and hit sequence are listed. The strand and orientation for query sequence and hits are also found here.

In most cases, the table view of the results will be easier to interpret than tens of sequence alignments.

12.6.7 I want to BLAST against my own sequence database, is this possible?

It is possible to download the entire BLAST program package and use it on your own computer, institution computer cluster or similar. This is preferred if you want to search in proprietary sequences or sequences unavailable in the public databases stored at NCBI. The downloadable BLAST package can either be installed as a web-based tool or as a command line tool. It is

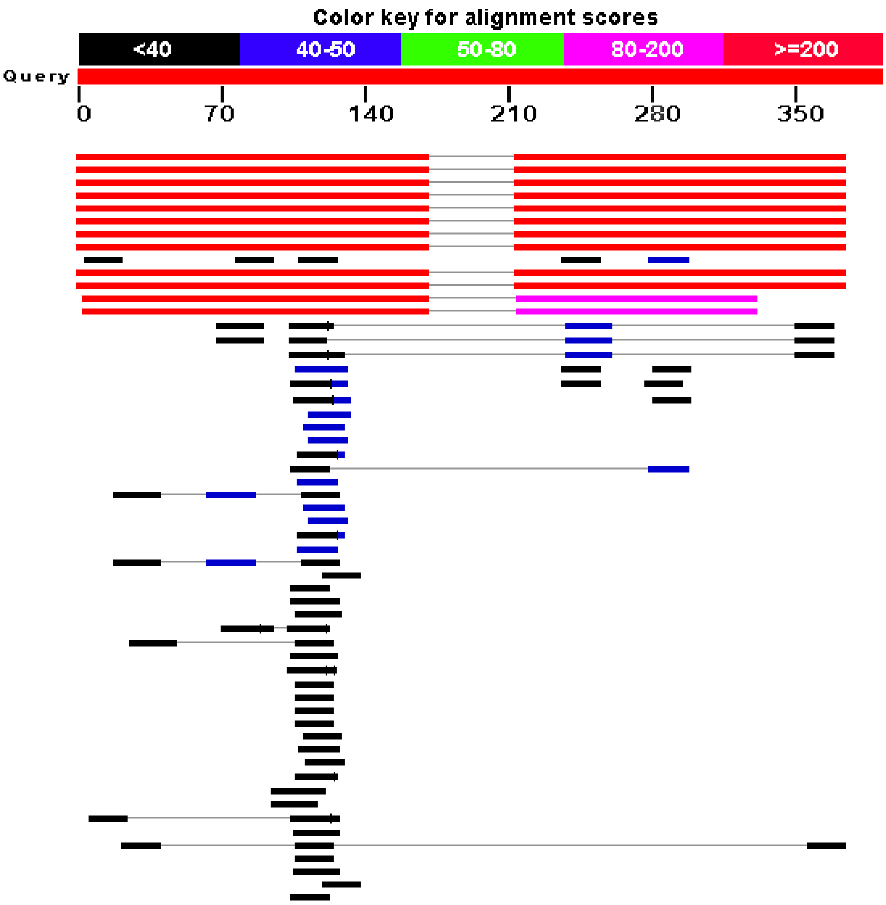
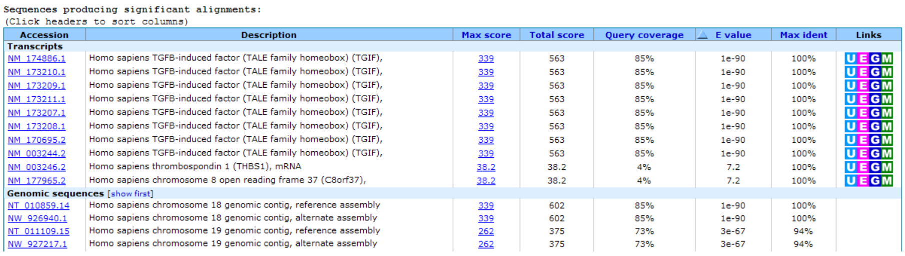


Figure 12.17: *BLAST graphical view. A simple graphical overview of the hits found aligned to the query sequence. The alignments are color coded ranging from black to red as indicated in the color label at the top.*



Sequences producing significant alignments:
(Click headers to sort columns)

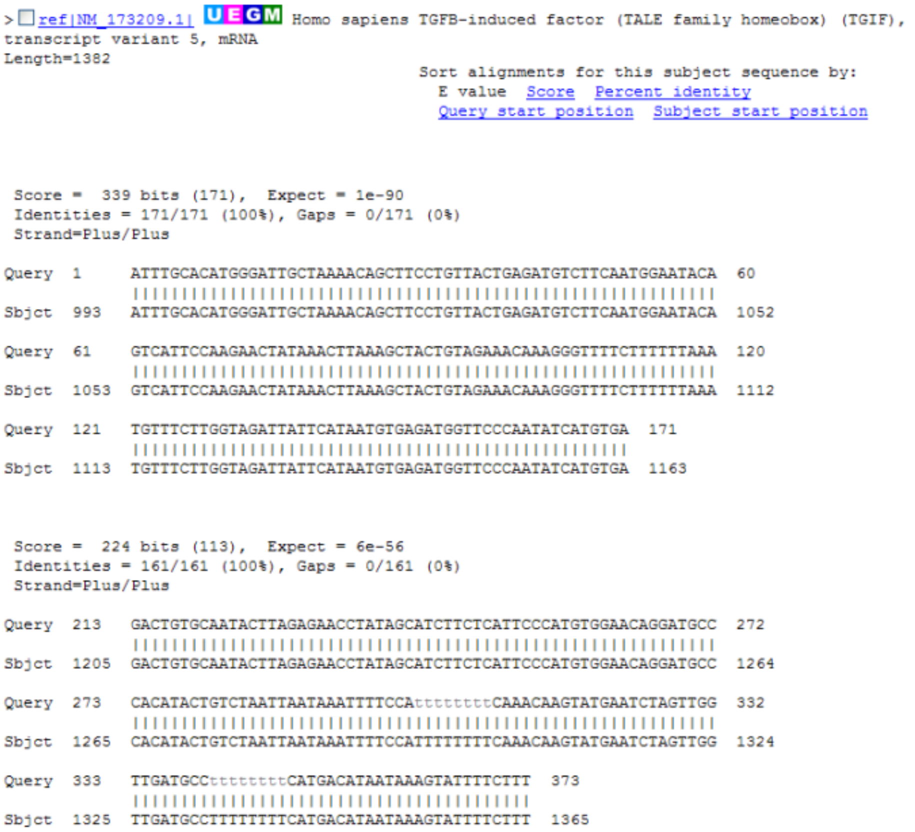
Accession	Description	Max score	Total score	Query coverage	E value	Max ident	Links
Transcripts							
NM_174886.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173210.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173209.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173211.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173207.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173208.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_170695.2	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_003244.2	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_003246.2	Homo sapiens thrombospondin 1 (THBS1), mRNA	38.2	38.2	4%	7.2	100%	U E G M
NM_177965.2	Homo sapiens chromosome 8 open reading frame 37 (C8orf37),	38.2	38.2	4%	7.2	100%	U E G M
Genomic sequences [show first]							
NT_010859.14	Homo sapiens chromosome 18 genomic contig, reference assembly	339	602	85%	1e-90	100%	
NW_926940.1	Homo sapiens chromosome 18 genomic contig, alternate assembly	339	602	85%	1e-90	100%	
NT_011109.15	Homo sapiens chromosome 19 genomic contig, reference assembly	262	375	73%	3e-67	94%	
NW_927217.1	Homo sapiens chromosome 19 genomic contig, alternate assembly	262	375	73%	3e-67	94%	

Figure 12.18: *BLAST table view. A table view with one row per hit, showing the accession number and description field from the sequence file together with BLAST output scores.*

available for a wide range of different operating systems.

The BLAST package can be downloaded free of charge from the following location http://www.ncbi.nlm.nih.gov/BLAST/download.shtml

Pre-formatted databases are available from a dedicated BLAST ftp site ftp://ftp.ncbi.nlm.nih.gov/blast/db/. Moreover, it is possible to download programs/scripts from the same site enabling automatic download of changed BLAST databases. Thus it is possible to schedule a nightly update of changed databases and have the updated BLAST database stored locally or on a shared network drive at all times. Most BLAST databases on the NCBI site are updated on

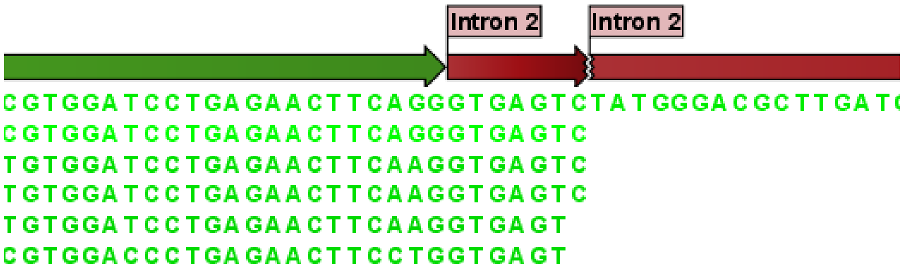


```
>□ref|NM_173209.1| UEGM Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),
transcript variant 5, mRNA
Length=1382
                                          Sort alignments for this subject sequence by:
                                            E value  Score  Percent identity
                                            Query start position  Subject start position


 Score =  339 bits (171),  Expect = 1e-90
 Identities = 171/171 (100%), Gaps = 0/171 (0%)
 Strand=Plus/Plus

Query  1     ATTTGCACATGGGATTGCTAAAACAGCTTCCTGTTACTGAGATGTCTTCAATGGAATACA  60
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  993   ATTTGCACATGGGATTGCTAAAACAGCTTCCTGTTACTGAGATGTCTTCAATGGAATACA  1052

Query  61    GTCATTCCAAGAACTATAAACTTAAAGCTACTGTAGAAACAAAGGGTTTTCTTTTTTAAA  120
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  1053  GTCATTCCAAGAACTATAAACTTAAAGCTACTGTAGAAACAAAGGGTTTTCTTTTTTAAA  1112

Query  121   TGTTTCTTGGTAGATTATTCATAATGTGAGATGGTTCCCAATATCATGTGA  171
             |||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  1113  TGTTTCTTGGTAGATTATTCATAATGTGAGATGGTTCCCAATATCATGTGA  1163


 Score =  224 bits (113),  Expect = 6e-56
 Identities = 161/161 (100%), Gaps = 0/161 (0%)
 Strand=Plus/Plus

Query  213   GACTGTGCAATACTTAGAGAACCTATAGCATCTTCTCATTCCCATGTGGAACAGGATGCC  272
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  1205  GACTGTGCAATACTTAGAGAACCTATAGCATCTTCTCATTCCCATGTGGAACAGGATGCC  1264

Query  273   CACATACTGTCTAATTAATAAATTTTCCAttttttttCAAACAAGTATGAATCTAGTTGG  332
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  1265  CACATACTGTCTAATTAATAAATTTTCCATTTTTTTTCAAACAAGTATGAATCTAGTTGG  1324

Query  333   TTGATGCCttttttttCATGACATAATAAAGTATTTTCTTT  373
             |||||||||||||||||||||||||||||||||||||||||
Sbjct  1325  TTGATGCCTTTTTTTTCATGACATAATAAAGTATTTTCTTT  1365
```

Figure 12.19: *Alignment view of BLAST results. Individual alignments are represented together with BLAST scores and more.*

a daily basis to include all recent sequence submissions to GenBank.

A few commercial software packages are available for searching your own data. The advantage of using a commercial program is obvious when BLAST is integrated with the existing tools of these programs. Furthermore, they let you perform BLAST searches and retain annotations on the query sequence (see figure 12.20). It is also much easier to batch download a selection of hit sequences for further inspection.



```
CGTGGATCCTGAGAACTTCAGGGTGAGTCTATGGGACGCTTGAT
CGTGGATCCTGAGAACTTCAGGGTGAGTC
TGTGGATCCTGAGAACTTCAAGGTGAGTC
TGTGGATCCTGAGAACTTCAAGGTGAGTC
TGTGGATCCTGAGAACTTCAAGGTGAGT
CGTGGACCCTGAGAACTTCCTGGTGAGT
```

Figure 12.20: *Snippet of alignment view of BLAST results from CLC Main Workbench. Individual alignments are represented directly in a graphical view. The top sequence is the query sequence and is shown with a selection of annotations.*

12.6.8 What you cannot get out of BLAST

Don't expect BLAST to produce the best available alignment. BLAST is a heuristic method which does not guarantee the best results, and therefore you cannot rely on BLAST if you wish to find *all* the hits in the database.

Instead, use the Smith-Waterman algorithm for obtaining the best possible local alignments [Smith and Waterman, 1981].

BLAST only makes local alignments. This means that a great but short hit in another sequence may not at all be related to the query sequence even though the sequences align well in a small region. It may be a domain or similar.

It is always a good idea to be cautious of the material in the database. For instance, the sequences may be wrongly annotated; hypothetical proteins are often simple translations of a found ORF on a sequenced nucleotide sequence and may not represent a true protein.

Don't expect to see the best result using the default settings. As described above, the settings should be adjusted according to the what kind of query sequence is used, and what kind of results you want. It is a good idea to perform the same BLAST search with different settings to get an idea of how they work. There is not a final answer on how to adjust the settings for your particular sequence.

12.6.9 Other useful resources

The BLAST web page hosted at NCBI
`http://www.ncbi.nlm.nih.gov/BLAST`

Download pages for the BLAST programs
`http://www.ncbi.nlm.nih.gov/BLAST/download.shtml`

Download pages for pre-formatted BLAST databases
`ftp://ftp.ncbi.nlm.nih.gov/blast/db/`

O'Reilly book on BLAST
`http://www.oreilly.com/catalog/blast/`

Explanation of scoring/substitution matrices and more
`http://www.clcbio.com/be/`

Creative Commons License

Chapter 13

3D molecule viewing

Contents

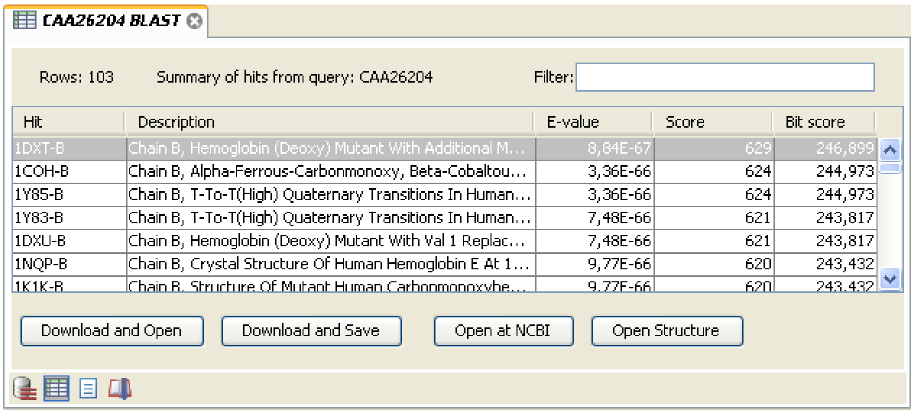
In order to understand protein function it is often valuable to see the actual three dimensional structure of the protein. This is of course only possible if the structure of the protein has been resolved and published. *CLC Genomics Workbench* has an integrated viewer of structure files. Structure files are usually deposited at the Protein DataBank (PDB) `www.rcsb.org`, where protein structure files can be searched and downloaded.

13.1 Importing structure files

In order to view the three dimensional structure files there are different ways to import these. The supported file formats are PDB and mmCIF which both can be downloaded from the Protein DataBank (`http://www.rcsb.org`) and imported through the import menu (see section 7.1.1).

Another way to import structure files is if a structure file is found through a direct search at the GenBank structure database (`http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Structure`). Read more about search for structures in section 11.3.

It is also possible to make a BLAST search against the PDB database. In the latter case, structure files can be directly downloaded to the navigation area by clicking the **Open structure** button below all the BLAST hits. Downloading structure files from a conducted BLAST search is only possible if the results are shown in a BLAST table. (See figure 13.1). How to conduct a BLAST search can be seen in section 12.1.



Rows: 103 Summary of hits from query: CAA26204 Filter:

Hit	Description	E-value	Score	Bit score
1DXT-B	Chain B, Hemoglobin (Deoxy) Mutant With Additional M...	8,84E-67	629	246,899
1COH-B	Chain B, Alpha-Ferrous-Carbonmonoxy, Beta-Cobaltou...	3,36E-66	624	244,973
1Y85-B	Chain B, T-To-T(High) Quaternary Transitions In Human...	3,36E-66	624	244,973
1Y83-B	Chain B, T-To-T(High) Quaternary Transitions In Human...	7,48E-66	621	243,817
1DXU-B	Chain B, Hemoglobin (Deoxy) Mutant With Val 1 Replac...	7,48E-66	621	243,817
1NQP-B	Chain B, Crystal Structure Of Human Hemoglobin E At 1...	9,77E-66	620	243,432
1K1K-B	Chain B, Structure Of Mutant Human Carbonmonoxyhe...	9,77E-66	620	243,432

Figure 13.1: *It is possible to open a structure file directly from the output of a conducted BLAST search by clicking the Open Structure button.*

13.2 Viewing structure files

An example of a 3D structure is shown in figure 13.2.

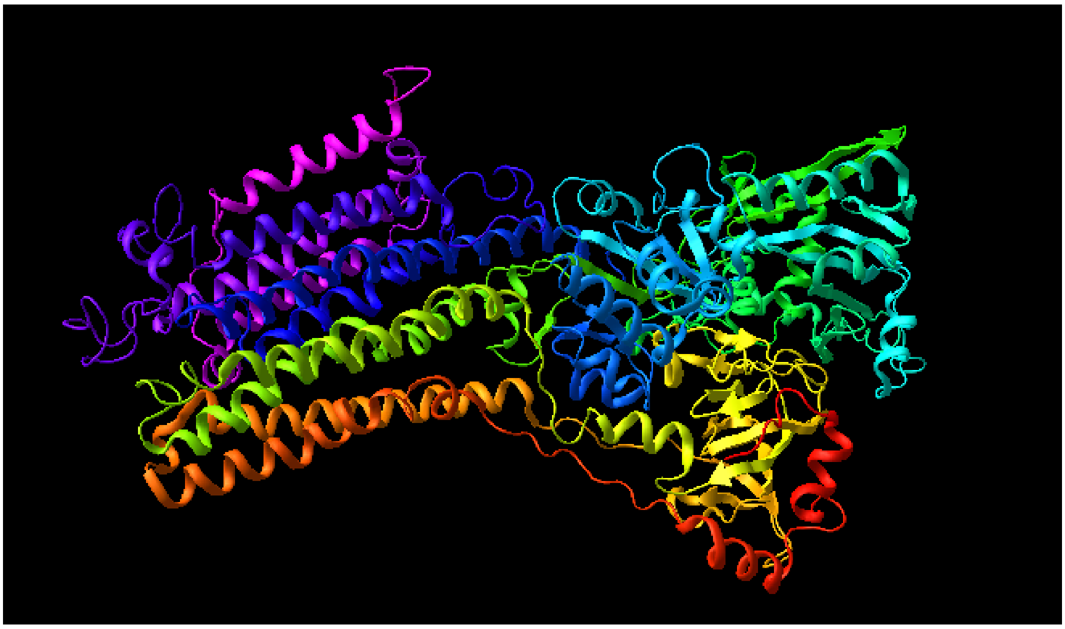


Figure 13.2: *3D view. Structure files can be opened, viewed and edited in several ways.*

Structures can be rotated and moved using the mouse and keyboard. **Pan mode** (✋) must be enabled in order to rotate and move the sequence.

Note! It is only possible to view one structure file at a time, in order to limit the amount of memory used.

13.2.1 Moving and rotating

Structure files are simply rotated by holding down the left mouse button while moving the mouse. This will rotate the structure in the direction the mouse is moved. The structures can be freely rotated in all directions.

Holding down the Ctrl on Windows or ⌘ on Mac key on the keyboard while dragging the mouse moves the structure in the direction the mouse is moved. This is particularly useful if the view is zoomed to cover only a small region of the protein structure.

Zoom in (🔍) and zoom out (🔍) on the structure is done by selecting the appropriate zoom tool in the toolbar and clicking with the mouse on the view area. Alternatively, click and hold the left mouse button while using either zoom tool and move the mouse up or down to zoom out or in respectively. The view can be restored to display the entire structure by clicking the **Fit width** ( ) button on the toolbar (read more about zooming in section 3.3).

- **Rotate mode**
  The structure is rotated when the "Pan mode" (✋) is selected in the toolbar. If the "pan mode" is not enabled on the first view of a structure a warning is shown.
- **Zoom mode**
  Use the zoom buttons on the toolbar to enable zoom mode. A single click with the mouse will zoom slightly on the structure. Moreover, it is possible to zoom in and out on the structure by keeping the left mouse button pressed while moving the mouse up and down.
- **Move mode**
  It is possible to move the structure from side to side if the Ctrl key on Windows and ⌘ key on Mac is pressed while dragging with the mouse.

13.3 Selections and display of the 3D structure

The view of the structure can be changed in several ways. All graphical changes are carried out through the **Side Panel**. At the top, you can change the default coloring (**Default colors** and **Settings**), and at the bottom you can change the representation of specific parts of the structure in order to high-light e.g. an active site.

13.3.1 Coloring of the 3D structure

The default colors apply if nothing else has been specified in the **Hierarchical view** below (see section 13.3.2):

- **Atom type**. Colors the atoms individually.
  - **Carbon**: Light grey
  - **Oxygen**: Red
  - **Hydrogen**: White
  - **Nitrogen**: Light blue
  - **Sulphur**: Yellow
  - **Chlorine, Boron**: Green
  - **Phosphorus, Iron, Barium**: Orange
  - **Sodium**: Blue
  - **Magnesium**: Forest green
  - **Zn, Cu, Ni, Br**: Brown
  - **Ca, Mn, Al, Ti, Cr, Ag**: Dark grey

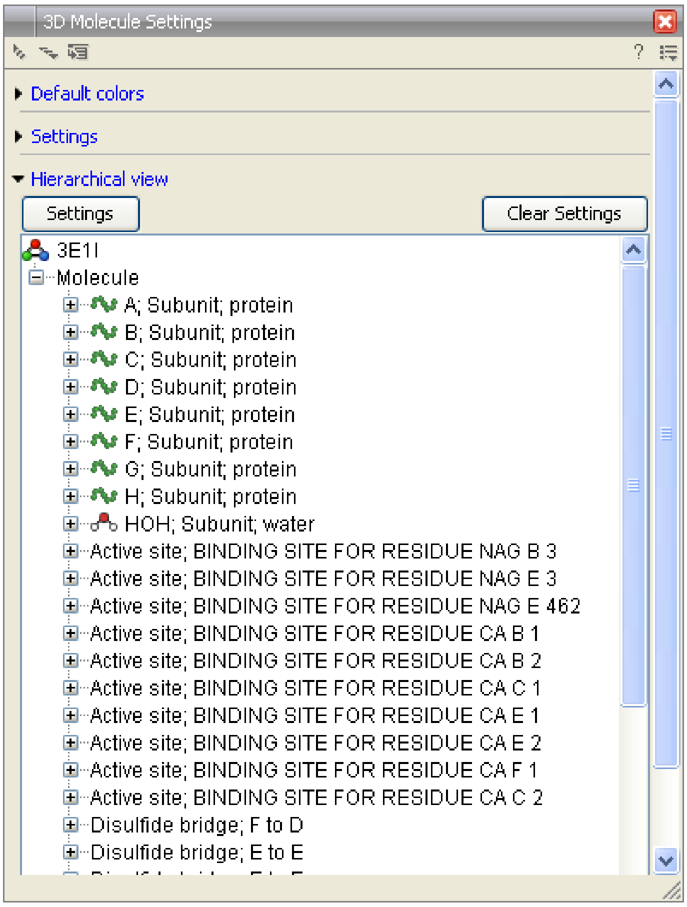
  - **F, Si, Au**: Goldenrod
  - **Iodine**: Purple
  - **Lithium**: firebrick
  - **Helium**: Pink
  - **Other**: Deep pink

- **Entities.** This will color protein subunits and additional structures individually. Using the view table, the user may select which colors are used to color subunits.
- **Rainbow.** This color mode will color the structure with rainbow colors along the sequence.
- **Residue hydrophobicity.** Colors the residues according to hydrophobicity.

In the **Settings** group, you can specify the background color to use. Default is black.

13.3.2 Hierarchical view - changing how selections of the structure are displayed

In the bottom of the **Side Panel**, you see the hierarchical view of the 3D structure (see an example in figure 13.3).

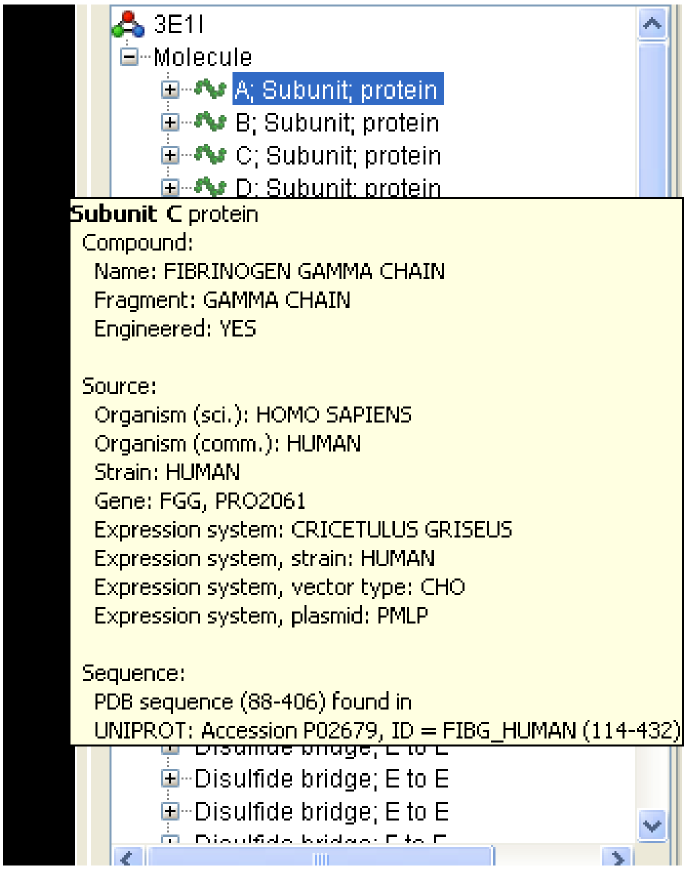


3E1I
Molecule
A; Subunit; protein
B; Subunit; protein
C; Subunit; protein
D; Subunit; protein
E; Subunit; protein
F; Subunit; protein
G; Subunit; protein
H; Subunit; protein
HOH; Subunit; water
Active site; BINDING SITE FOR RESIDUE NAG B 3
Active site; BINDING SITE FOR RESIDUE NAG E 3
Active site; BINDING SITE FOR RESIDUE NAG E 462
Active site; BINDING SITE FOR RESIDUE CA B 1
Active site; BINDING SITE FOR RESIDUE CA B 2
Active site; BINDING SITE FOR RESIDUE CA C 1
Active site; BINDING SITE FOR RESIDUE CA E 1
Active site; BINDING SITE FOR RESIDUE CA E 2
Active site; BINDING SITE FOR RESIDUE CA F 1
Active site; BINDING SITE FOR RESIDUE CA C 2
Disulfide bridge; F to D
Disulfide bridge; E to E

Figure 13.3: *Hierarchy view in the Side Panel.*

The hierarchical view shows the structure in a detailed manner. Individual structure subunits, residues, active sites, disulfide bridges or even down to the atom level can be selected individually and colored accordingly.

You can show additional information from the hierarchical view by holding your mouse still for one second (see an example in figure 13.4).



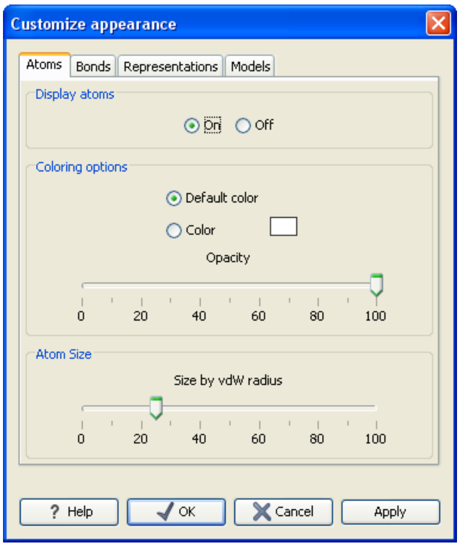
Subunit C protein
Compound:
Name: FIBRINOGEN GAMMA CHAIN
Fragment: GAMMA CHAIN
Engineered: YES

Source:
Organism (sci.): HOMO SAPIENS
Organism (comm.): HUMAN
Strain: HUMAN
Gene: FGG, PRO2061
Expression system: CRICETULUS GRISEUS
Expression system, strain: HUMAN
Expression system, vector type: CHO
Expression system, plasmid: PMLP

Sequence:
PDB sequence (88-406) found in
UNIPROT: Accession P02679, ID = FIBG_HUMAN (114-432)

Figure 13.4: *Details shown by holding the mouse cursor on a subunit.*

For each item in the hierarchy view, you can apply special view settings. Simply click an item in the hierarchy view and click the **Settings** button above. This will display the settings dialog as shown in figure 13.5.



Display atoms
● On ○ Off

Coloring options
● Default color
○ Color
Opacity

Atom Size
Size by vdW radius

Figure 13.5: *Customize appearance.*

There are four tabs at the top. You can specify settings for each tab and then click **OK** to apply and close the dialog. You can also click **Apply** which will keep the settings dialog visible. You will then be able to select another item in the hierarchical view and apply settings for this also.

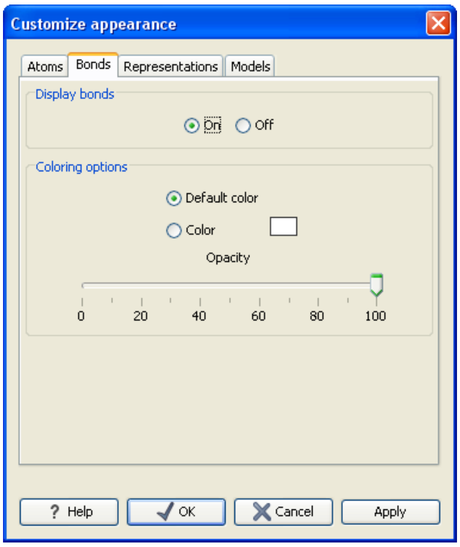
Atoms

The **Atoms** tab is shown in figure 13.5. At the top, you can choose to show atoms, and below you can specify their appearance:

- **Color**. Clicking the color box allows you to select a color.
- **Opacity**. Determines the level of opacity.
- **Atom size**. The size of the atoms measured by vdW radius.

Bonds

The **Bonds** tab is shown in figure 13.6.


Customize appearance

Display bonds

● On ○ Off

Coloring options

● Default color

○ Color □

Opacity

Figure 13.6: *Customize appearance for bonds.*

At the top, you can choose to show bonds, and below you can specify their appearance:

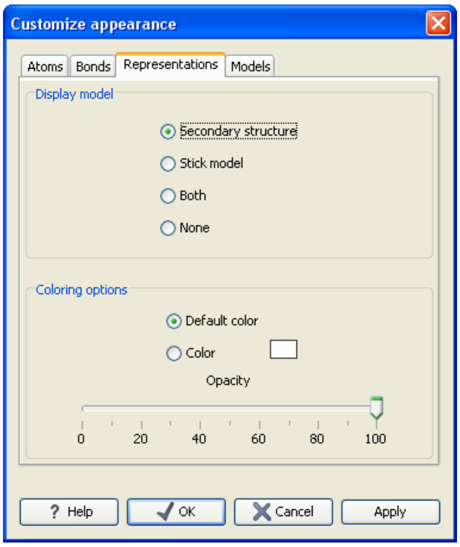
- **Color**. Clicking the color box allows you to select a color.
- **Opacity**. Determines the level of opacity.

Representations

The **Representations** tab is shown in figure 13.7.

At the top, you can choose to between four display models:

- **Secondary structure**.



Display model: ● Secondary structure ○ Stick model ○ Both ○ None

Coloring options: ● Default color ○ Color □ Opacity: 100

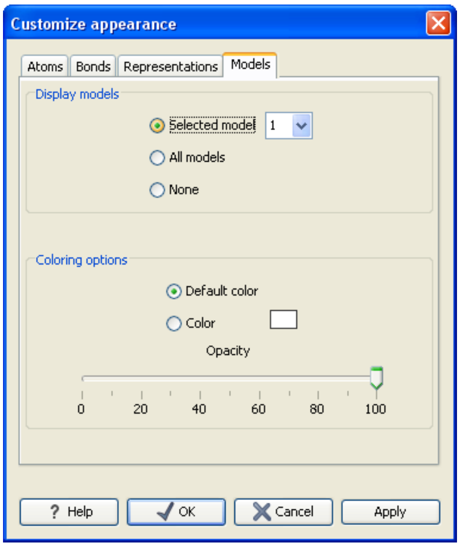
Figure 13.7: *Customize appearance for representations.*

- **Stick model**.
- **Both**. Displaying both secondary structure and stick model.
- **None**. Will not display representations.

- **Color**. Clicking the color box allows you to select a color.
- **Opacity**. Determines the level of opacity.

Models

The **Models** tab is shown in figure 13.8. At the top, you can choose to between three display modes. This functionality is only applicable to NMR structures which have multiple resolved structures, for X-ray structures only one structure is available:



Display models: ● Selected model 1 ○ All models ○ None

Coloring options: ● Default color ○ Color □ Opacity: 100

Figure 13.8: *Customize appearance for models.*

- **Selected model**.
- **All models**.
- **None**. Will not display models.

- **Color**. Clicking the color box allows you to select a color.
- **Opacity**. Determines the level of opacity.

13.4 3D Output

The output of the 3D view is rendered on the screen in real time and changes to the preferences are visible immediately. From *CLC Genomics Workbench* you can export the visible part of the 3D view to different graphic formats, by pressing the **Graphics** button (🖼) on the **Menu bar**. This will allow you to export in the following formats:

Format	Suffix	Type
Portable Network Graphics	.png	bitmap
JPEG	.jpg	bitmap
Tagged Image File	.tif	bitmap
PostScript	.ps	vector graphics
Encapsulated PostScript	.eps	vector graphics
Portable Document Format	.pdf	vector graphics
Scalable Vector Graphics	.svg	vector graphics

Printing is not fully implemented with the 3D editor. Should you wish to print a 3D view, this can be done by either exporting to a graphics format and printing that or use the scheme below.

Windows:

- Adjust your 3D view in *CLC Genomics Workbench*
- Press **Print Screen** on your keyboard (or **Alt + Print Screen**)
- Paste the result into an 'image editor' e.g. Paint or GIMP http://www.gimp.org/
- Crop (edit the screen shot)
- Save in your preferred file format and/or print

Mac:

- Set up your 3D view
- Press ⌘ **+ shift + 3** (or ⌘ **+ shift + 4**) (to take screen shot)
- Open the saved file (.pdf or .png) in a 'image editor' e.g. GIMP http://www.gimp.org/
- Crop (edit the screen shot)
- Save in your preferred file format and/or print

Linux:

- Set up your 3D view
- e.g. use GIMP to take the screen shot http://www.gimp.org/
- Crop (edit the screen shot)
- Save in your preferred file format and/or print

Chapter 14

General sequence analyses

Contents

CLC Genomics Workbench offers different kinds of sequence analyses, which apply to both protein and DNA. The analyses are described in this chapter.

14.1 Shuffle sequence

In some cases, it is beneficial to shuffle a sequence. This is an option in the **Toolbox** menu under **General Sequence Analyses**. It is normally used for statistical analyses, e.g. when comparing an alignment score with the distribution of scores of shuffled sequences.

Shuffling a sequence removes all annotations that relate to the residues.

> **select sequence | Toolbox in the Menu Bar | General Sequence Analyses ( ) | Shuffle Sequence ( )**

or **right-click a sequence** | **Toolbox** | **General Sequence Analyses** **()** | **Shuffle Sequence** **()**

This opens the dialog displayed in figure 14.1:

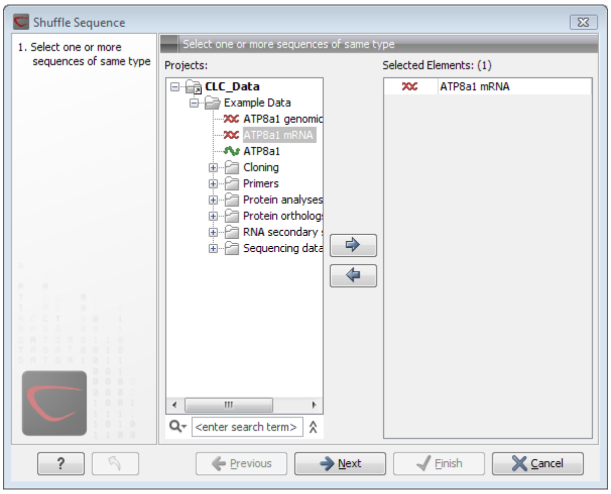
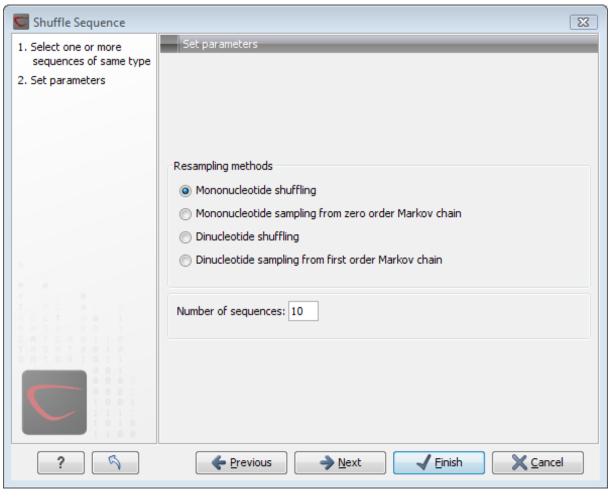


Figure 14.1: *Choosing sequence for shuffling.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** to determine how the shuffling should be performed.

In this step, shown in figure 14.2: For nucleotides, the following parameters can be set:



Resampling methods

● Mononucleotide shuffling

○ Mononucleotide sampling from zero order Markov chain

○ Dinucleotide shuffling

○ Dinucleotide sampling from first order Markov chain

Number of sequences: 10

Figure 14.2: *Parameters for shuffling.*

- **Mononucleotide shuffling.** Shuffle method generating a sequence of the exact same mononucleotide frequency
- **Dinucleotide shuffling.** Shuffle method generating a sequence of the exact same dinucleotide frequency
- **Mononucleotide sampling from zero order Markov chain.** Resampling method generating a sequence of the same expected mononucleotide frequency.

- **Dinucleotide sampling from first order Markov chain.** Resampling method generating a sequence of the same expected dinucleotide frequency.

For proteins, the following parameters can be set:

- **Single amino acid shuffling.** Shuffle method generating a sequence of the exact same amino acid frequency.
- **Single amino acid sampling from zero order Markov chain.** Resampling method generating a sequence of the same expected single amino acid frequency.
- **Dipeptide shuffling.** Shuffle method generating a sequence of the exact same dipeptide frequency.
- **Dipeptide sampling from first order Markov chain.** Resampling method generating a sequence of the same expected dipeptide frequency.

For further details of these algorithms, see [Clote et al., 2005]. In addition to the shuffle method, you can specify the number of randomized sequences to output.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

This will open a new view in the **View Area** displaying the shuffled sequence. The new sequence is not saved automatically. To save the sequence, drag it into the **Navigation Area** or press ctrl + S (⌘ + S on Mac) to activate a save dialog.

14.2 Dot plots

Dot plots provide a powerful visual comparison of two sequences. Dot plots can also be used to compare regions of similarity within a sequence. This chapter first describes how to create and second how to adjust the view of the plot.

14.2.1 Create dot plots

A dot plot is a simple, yet intuitive way of comparing two sequences, either DNA or protein, and is probably the oldest way of comparing two sequences [Maizel and Lenk, 1981]. A dot plot is a 2 dimensional matrix where each axis of the plot represents one sequence. By sliding a fixed size window over the sequences and making a sequence match by a dot in the matrix, a diagonal line will emerge if two identical (or very homologous) sequences are plotted against each other. Dot plots can also be used to visually inspect sequences for direct or inverted repeats or regions with low sequence complexity. Various smoothing algorithms can be applied to the dot plot calculation to avoid noisy background of the plot. Moreover, can various substitution matrices be applied in order to take the evolutionary distance of the two sequences into account.

To create a dot plot:

Toolbox | **General Sequence Analyses** ( ) | **Create Dot Plot** ( )

or **Select one or two sequences in the Navigation Area** | **Toolbox in the Menu Bar** | **General Sequence Analyses** ( ) | **Create Dot Plot** ( )

or **Select one or two sequences in the Navigation Area | right-click in the Navigation Area | Toolbox | General Sequence Analyses ( ) | Create Dot Plot ( )**

This opens the dialog shown in figure 14.3.

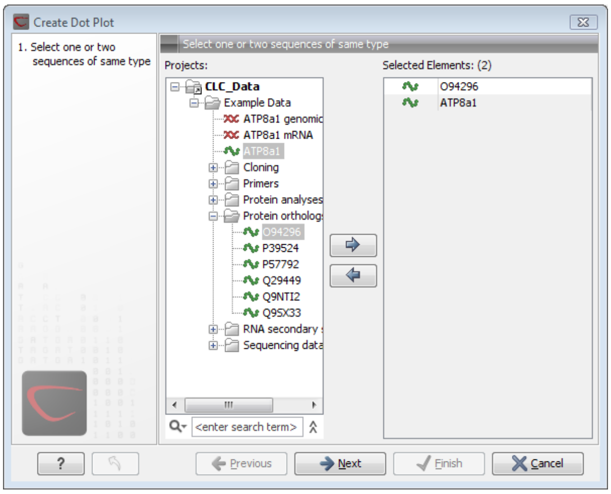


Figure 14.3: *Selecting sequences for the dot plot.*

If a sequence was selected before choosing the **Toolbox** action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove elements from the selected elements. Click **Next** to adjust dot plot parameters. Clicking **Next** opens the dialog shown in figure 14.4.

Notice! Calculating dot plots take up a considerable amount of memory in the computer. Therefore, you see a warning if the sum of the number of nucleotides/amino acids in the sequences is higher than 8000. If you insist on calculating a dot plot with more residues the Workbench may shut down, allowing you to save your work first. However, this depends on your computer's memory configuration.

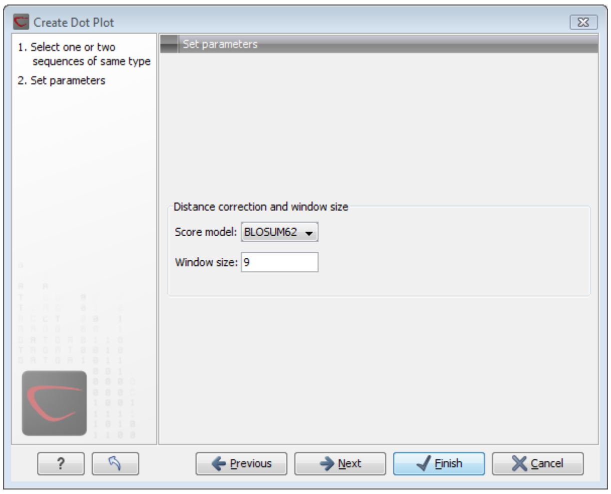
Adjust dot plot parameters

There are two parameters for calculating the dot plot:

- **Distance correction (only valid for protein sequences)** In order to treat evolutionary transitions of amino acids, a distance correction measure can be used when calculating the dot plot. These distance correction matrices (substitution matrices) take into account the likeliness of one amino acid changing to another.

- **Window size** A residue by residue comparison (window size = 1) would undoubtedly result in a very noisy background due to a lot of similarities between the two sequences of interest. For DNA sequences the background noise will be even more dominant as a match between only four nucleotide is very likely to happen. Moreover, a residue by residue comparison (window size = 1) can be very time consuming and computationally demanding. Increasing the window size will make the dot plot more 'smooth'.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.



Create Dot Plot

1. Select one or two sequences of same type
2. Set parameters

Set parameters

Distance correction and window size

Score model: BLOSUM62

Window size: 9

Figure 14.4: *Setting the dot plot parameters.*

14.2.2 View dot plots

A view of a dot plot can be seen in figure 14.5. You can select **Zoom in** () in the Toolbar and click the dot plot to zoom in to see the details of particular areas.

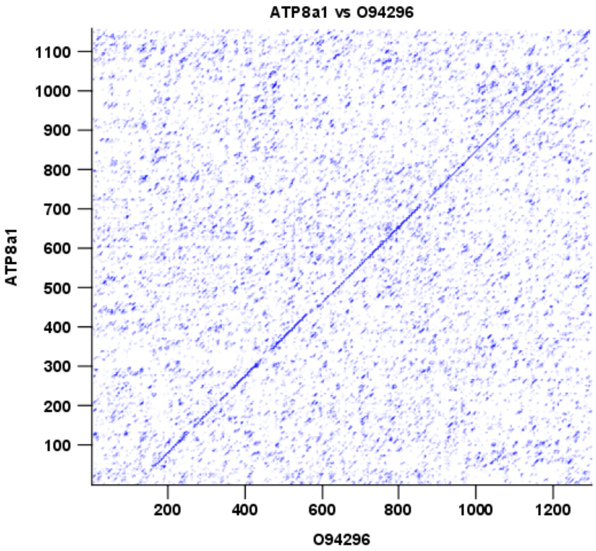


Figure 14.5: *A view is opened showing the dot plot.*

The **Side Panel** to the right let you specify the dot plot preferences. The gradient color box can be adjusted to get the appropriate result by dragging the small pointers at the top of the box. Moving the slider from the right to the left lowers the thresholds which can be directly seen in the dot plot, where more diagonal lines will emerge. You can also choose another color gradient by clicking on the gradient box and choose from the list.

Adjusting the sliders above the gradient box is also practical, when producing an output for printing. (Too much background color might not be desirable). By crossing one slider over the other (the two sliders change side) the colors are inverted, allowing for a white background. (If you choose a color gradient, which includes white). Se figure 14.5.

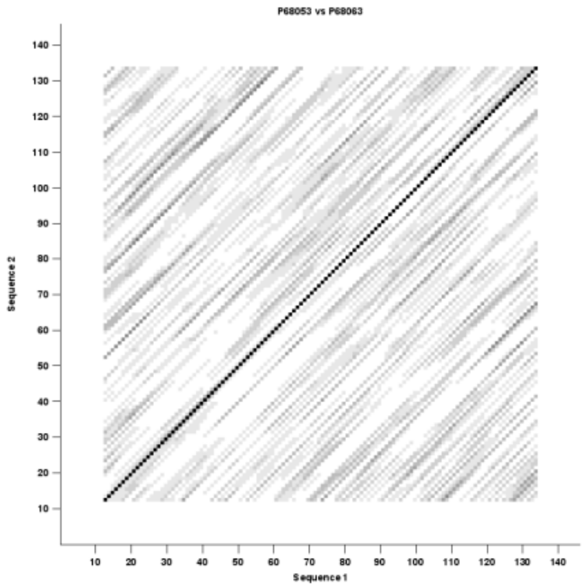


Figure 14.6: *Dot plot with inverted colors, practical for printing.*

14.2.3 Bioinformatics explained: Dot plots

Realization of dot plots

Dot plots are two-dimensional plots where the x-axis and y-axis each represents a sequence and the plot itself shows a comparison of these two sequences by a calculated score for each position of the sequence. If a window of fixed size on one sequence (one axis) match to the other sequence a dot is drawn at the plot. Dot plots are one of the oldest methods for comparing two sequences [Maizel and Lenk, 1981].

The scores that are drawn on the plot are affected by several issues.

- Scoring matrix for distance correction.
  Scoring matrices (BLOSUM and PAM) contain substitution scores for every combination of two amino acids. Thus, these matrices can only be used for dot plots of protein sequences.

- Window size
  The single residue comparison (bit by bit comparison(window size = 1)) in dot plots will undoubtedly result in a noisy background of the plot. You can imagine that there are many successes in the comparison if you only have four possible residues like in nucleotide sequences. Therefore you can set a window size which is smoothing the dot plot. Instead of comparing single residues it compares subsequences of length set as window size. The score is now calculated with respect to aligning the subsequences.

- Threshold
  The dot plot shows the calculated scores with colored threshold. Hence you can better recognize the most important similarities.

Examples and interpretations of dot plots

Contrary to simple sequence alignments dot plots can be a very useful tool for spotting various evolutionary events which may have happened to the sequences of interest.

Below is shown some examples of dot plots where sequence insertions, low complexity regions, inverted repeats etc. can be identified visually.

Similar sequences

The most simple example of a dot plot is obtained by plotting two homologous sequences of interest. If very similar or identical sequences are plotted against each other a diagonal line will occur.

The dot plot in figure 14.7 shows two related sequences of the Influenza A virus nucleoproteins infecting ducks and chickens. Accession numbers from the two sequences are: DQ232610 and DQ023146. Both sequences can be retrieved directly from http://www.ncbi.nlm.nih.gov/gquery/gquery.fcgi.

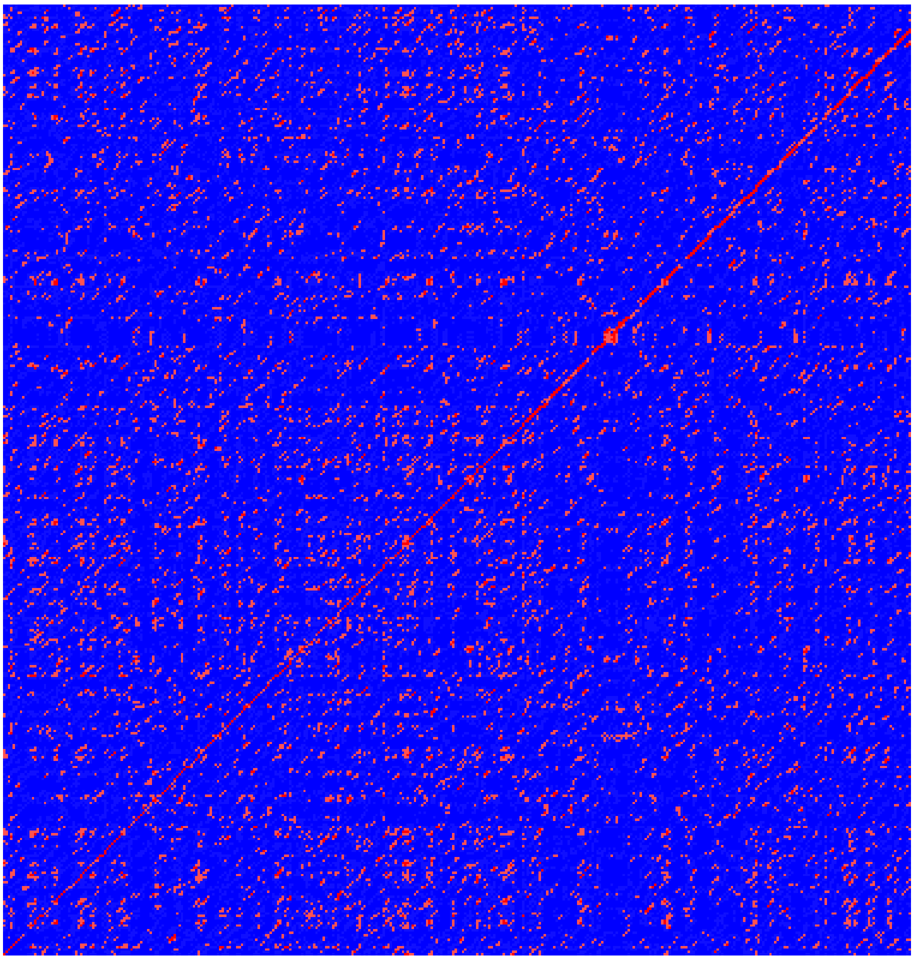


Figure 14.7: *Dot plot of DQ232610 vs. DQ023146 (Influenza A virus nucleoproteins) showing and overall similarity*

Repeated regions

Sequence repeats can also be identified using dot plots. A repeat region will typically show up as lines parallel to the diagonal line.

If the dot plot shows more than one diagonal in the same region of a sequence, the regions depending to the other sequence are repeated. In figure 14.9 you can see a sequence with repeats.

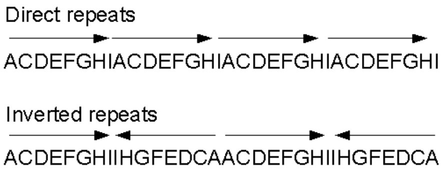

Direct repeats
ACDEFGHIACDEFGHIACDEFGHIACDEFGHI
Inverted repeats
ACDEFGHIIHGFEDCAACDEFGHIIHGFEDCA

Figure 14.8: *Direct and inverted repeats shown on an amino acid sequence generated for demonstration purposes.*

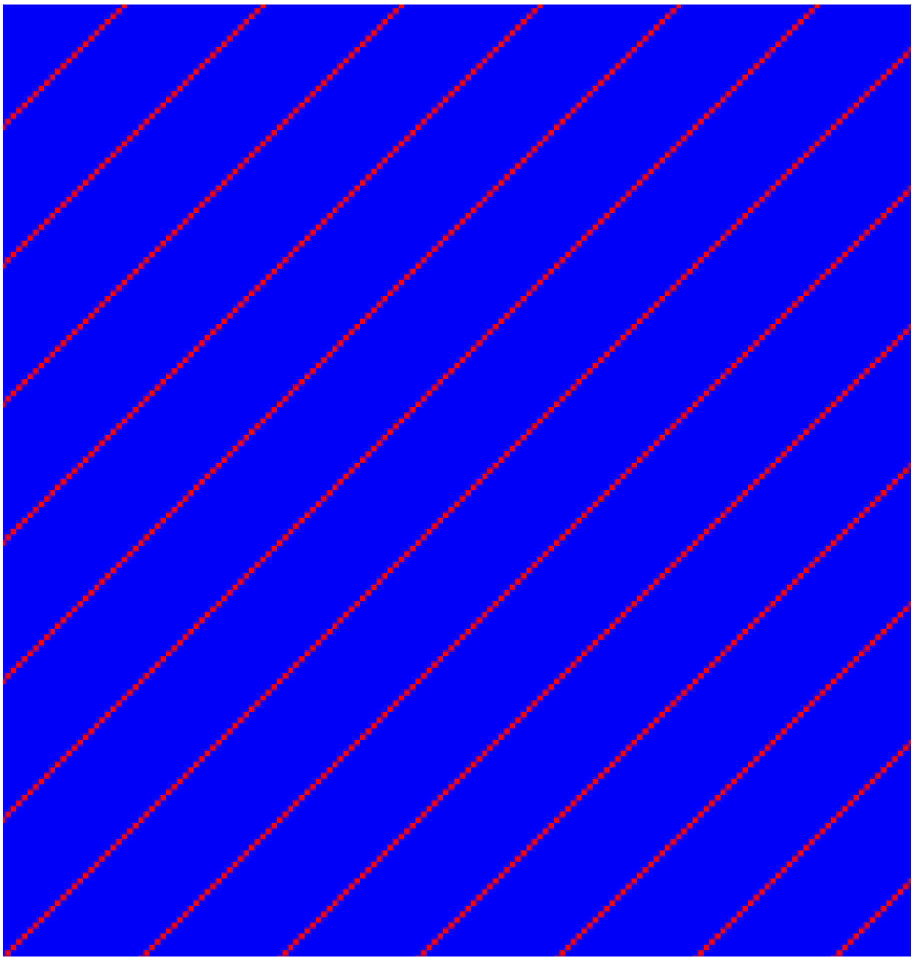


Figure 14.9: *The dot plot of a sequence showing repeated elements. See also figure 14.8.*

Frame shifts

Frame shifts in a nucleotide sequence can occur due to insertions, deletions or mutations. Such frame shifts can be visualized in a dot plot as seen in figure 14.10. In this figure, three frame shifts for the sequence on the y-axis are found.

1. Deletion of nucleotides
2. Insertion of nucleotides
3. Mutation (out of frame)

Sequence inversions

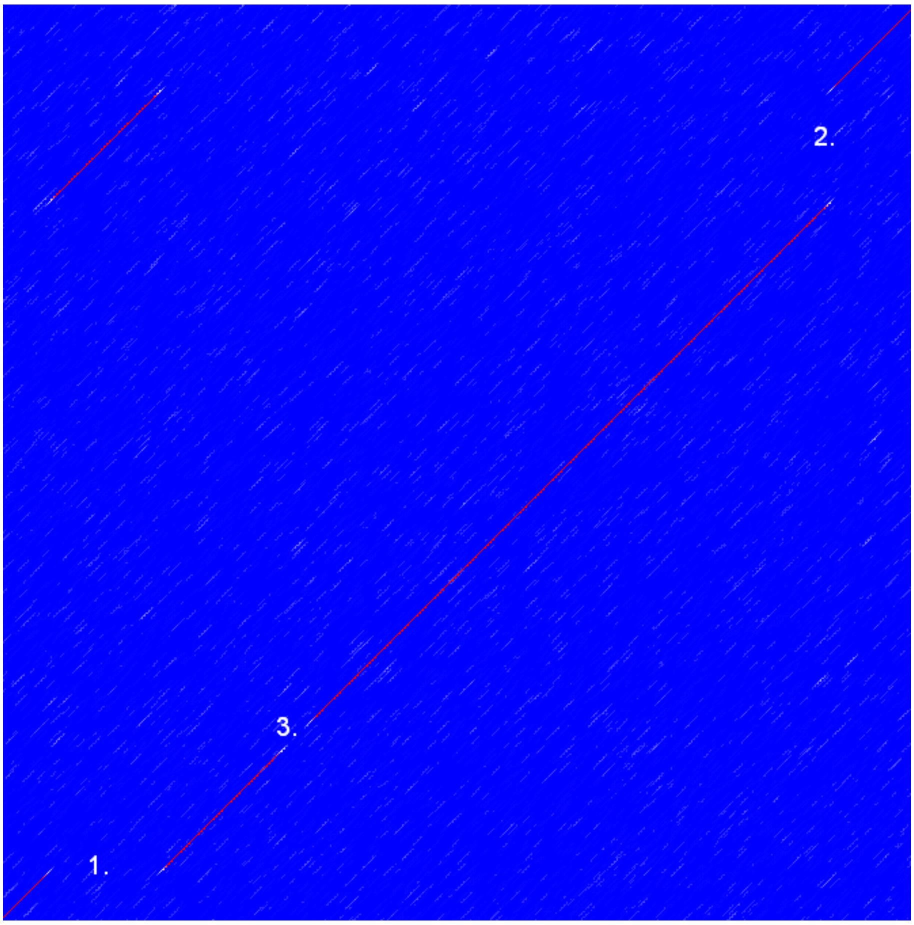


Figure 14.10: *This dot plot show various frame shifts in the sequence. See text for details.*

In dot plots you can see an inversion of sequence as contrary diagonal to the diagonal showing similarity. In figure 14.11 you can see a dot plot (window length is 3) with an inversion.

Low-complexity regions

Low-complexity regions in sequences can be found as regions around the diagonal all obtaining a high score. Low complexity regions are calculated from the redundancy of amino acids within a limited region [Wootton and Federhen, 1993]. These are most often seen as short regions of only a few different amino acids. In the middle of figure 14.12 is a square shows the low-complexity region of this sequence.

Creative Commons License

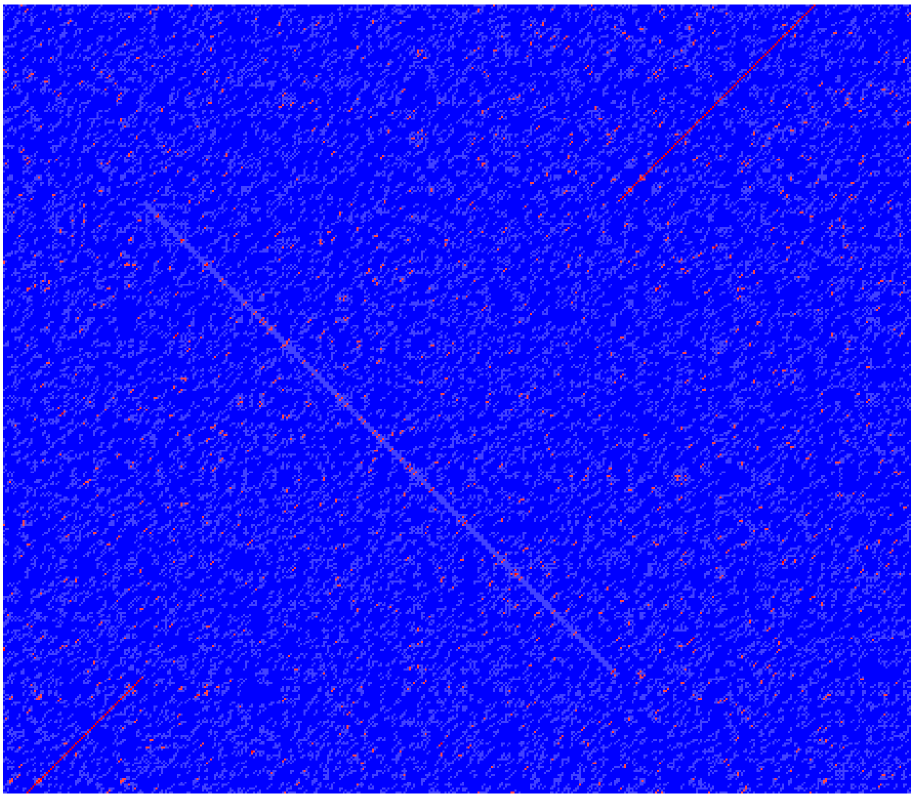


Figure 14.11: *The dot plot showing a inversion in a sequence. See also figure 14.8.*

14.2.4 Bioinformatics explained: Scoring matrices

Biological sequences have evolved throughout time and evolution has shown that not all changes to a biological sequence is equally likely to happen. Certain amino acid substitutions (change of one amino acid to another) happen often, whereas other substitutions are very rare. For instance, tryptophan (W) which is a relatively rare amino acid, will only — on very rare occasions — mutate into a leucine (L).

Based on evolution of proteins it became apparent that these changes or substitutions of amino acids can be modeled by a scoring matrix also refereed to as a substitution matrix. See an example of a scoring matrix in table 14.1. This matrix lists the substitution scores of every single amino acid. A score for an aligned amino acid pair is found at the intersection of the corresponding column and row. For example, the substitution score from an arginine (R) to a lysine (K) is 2. The diagonal show scores for amino acids which have not changed. Most substitutions changes have a negative score. Only rounded numbers are found in this matrix.

The two most used matrices are the BLOSUM [Henikoff and Henikoff, 1992] and PAM [Dayhoff and Schwartz, 1978].

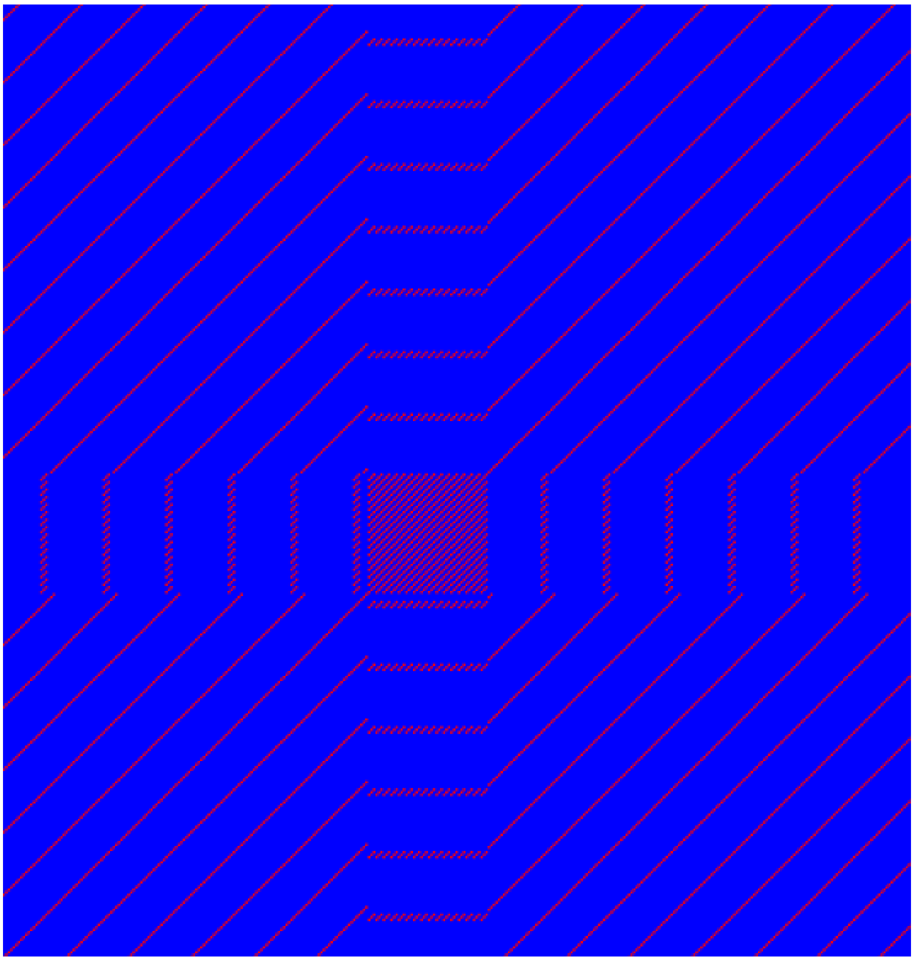


Figure 14.12: *The dot plot showing a low-complexity region in the sequence. The sequence is artificial and low complexity regions does not always show as a square.*

Different scoring matrices

PAM

The first PAM matrix (Point Accepted Mutation) was published in 1978 by Dayhoff et al. The PAM matrix was build through a global alignment of related sequences all having sequence similarity above 85% [Dayhoff and Schwartz, 1978]. A PAM matrix shows the probability that any given amino acid will mutate into another in a given time interval. As an example, PAM1 gives that one amino acid out of a 100 will mutate in a given time interval. In the other end of the scale, a PAM256 matrix, gives the probability of 256 mutations in a 100 amino acids (see figure 14.13).

There are some limitation to the PAM matrices which makes the BLOSUM matrices somewhat more attractive. The dataset on which the initial PAM matrices were build is very old by now, and the PAM matrices assume that all amino acids mutate at the same rate - this is not a correct assumption.

BLOSUM

In 1992, 14 years after the PAM matrices were published, the BLOSUM matrices (BLOcks SUbstitution Matrix) were developed and published [Henikoff and Henikoff, 1992].

Henikoff et al. wanted to model more divergent proteins, thus they used locally aligned sequences where none of the aligned sequences share less than 62% identity. This resulted

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Table 14.1: **The BLOSUM62 matrix.** A tabular view of the BLOSUM62 matrix containing all possible substitution scores [Henikoff and Henikoff, 1992].

in a scoring matrix´called BLOSUM62. In contrast to the PAM matrices the BLOSUM matrices are calculated from alignments without gaps emerging from the BLOCKS database http://blocks.fhcrc.org/.

Sean Eddy recently wrote a paper reviewing the BLOSUM62 substitution matrix and how to calculate the scores [Eddy, 2004].

Use of scoring matrices

Deciding which scoring matrix you should use in order of obtain the best alignment results is a difficult task. If you have no prior knowledge on the sequence the BLOSUM62 is probably the best choice. This matrix has become the *de facto* standard for scoring matrices and is also used as the default matrix in BLAST searches. The selection of a "wrong" scoring matrix will most probable strongly influence on the outcome of the analysis. In general a few rules apply to the selection of scoring matrices.

- For closely related sequences choose BLOSUM matrices created for highly similar alignments, like BLOSUM80. You can also select low PAM matrices such as PAM1.
- For distant related sequences, select low BLOSUM matrices (for example BLOSUM45) or high PAM matrices such as PAM250.

The BLOSUM matrices with low numbers correspond to PAM matrices with high numbers. (See figure 14.13) for correlations between the PAM and BLOSUM matrices. To summarize, if you want to find distant related proteins to a sequence of interest using BLAST, you could benefit of using BLOSUM45 or similar matrices.

Other useful resources

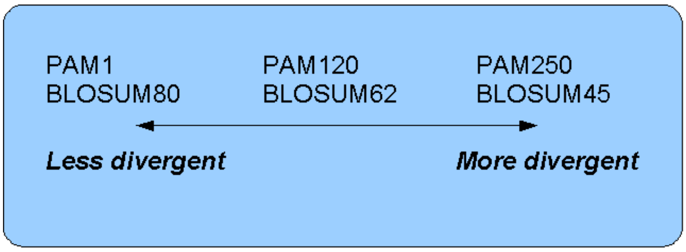


Figure 14.13: *Relationship between scoring matrices. The BLOSUM62 has become a* de facto *standard scoring matrix for a wide range of alignment programs. It is the default matrix in BLAST.*

Calculate your own PAM matrix
http://www.bioinformatics.nl/tools/pam.html

BLOKS database
http://blocks.fhcrc.org/

NCBI help site
http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/Scoring2.html

Creative Commons License

14.3 Local complexity plot

In *CLC Genomics Workbench* it is possible to calculate local complexity for both DNA and protein sequences. The local complexity is a measure of the diversity in the composition of amino acids within a given range (window) of the sequence. The K2 algorithm is used for calculating local complexity [Wootton and Federhen, 1993]. To conduct a complexity calculation do the following:

> **Select sequences in Navigation Area | Toolbox in Menu Bar | General Sequence Analyses ( ) | Create Complexity Plot ( )**

This opens a dialog. In **Step 1** you can change, remove and add DNA and protein sequences.

When the relevant sequences are selected, clicking **Next** takes you to **Step 2**. This step allows you to adjust the window size from which the complexity plot is calculated. Default is set to 11 amino acids and the number should always be odd. The higher the number, the less volatile the graph.

Figure 14.14 shows an example of a local complexity plot.

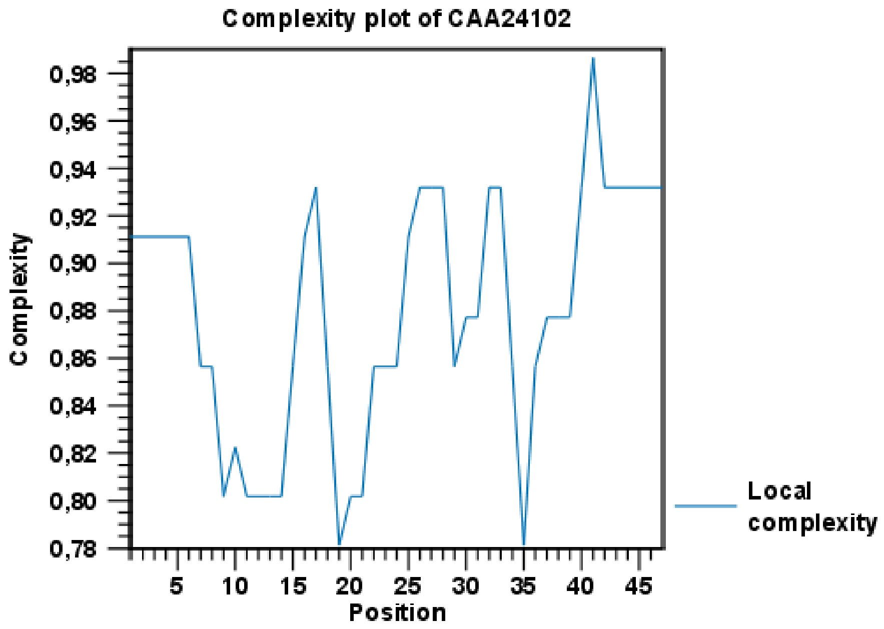


Figure 14.14: *An example of a local complexity plot.*

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The values of the complexity plot approaches 1.0 as the distribution of amino acids become more complex.

See section B in the appendix for information about the graph view.

14.4 Sequence statistics

CLC Genomics Workbench can produce an output with many relevant statistics for protein sequences. Some of the statistics are also relevant to produce for DNA sequences. Therefore, this section deals with both types of statistics. The required steps for producing the statistics are the same.

To create a statistic for the sequence, do the following:

> **select sequence(s)** | **Toolbox in the Menu Bar** | **General Sequence Analyses ( )** | **Create Sequence Statistics ( )**

This opens a dialog where you can alter your choice of sequences which you want to create statistics for. You can also add sequence lists.

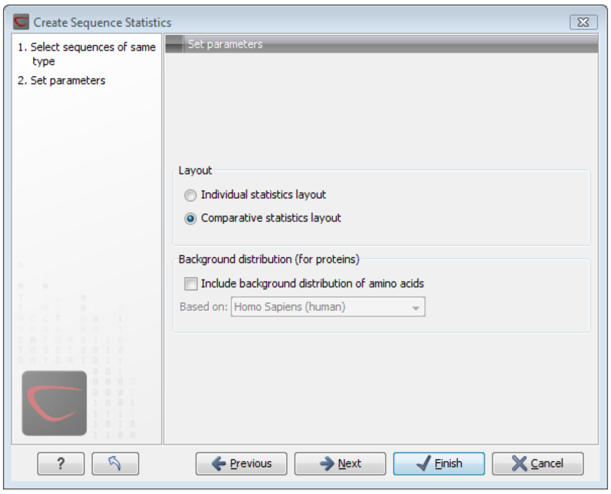
Note! You cannot create statistics for DNA and protein sequences at the same time.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 14.15.

The dialog offers to adjust the following parameters:

- **Individual statistics layout.** If more sequences were selected in **Step 1**, this function generates separate statistics for each sequence.
- **Comparative statistics layout.** If more sequences were selected in **Step 1**, this function



Layout

○ Individual statistics layout

● Comparative statistics layout

Background distribution (for proteins)

□ Include background distribution of amino acids

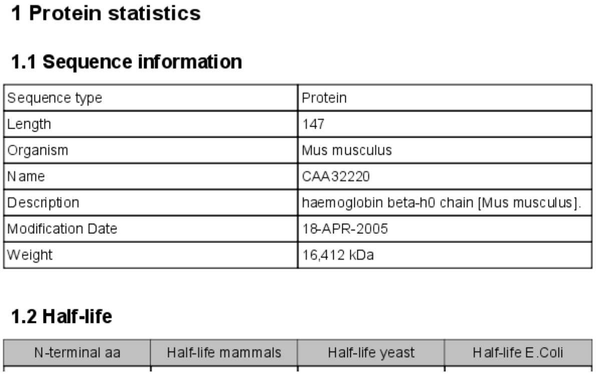
Based on: Homo Sapiens (human)

Figure 14.15: *Setting parameters for the sequence statistics.*

generates statistics with comparisons between the sequences.

You can also choose to include Background distribution of amino acids. If this box is ticked, an extra column with amino acid distribution of the chosen species, is included in the table output. (The distributions are calculated from UniProt `www.uniprot.org` version 6.0, dated September 13 2005.)

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. An example of protein sequence statistics is shown in figure 14.16.



1 Protein statistics

1.1 Sequence information

Sequence type	Protein
Length	147
Organism	Mus musculus
Name	CAA32220
Description	haemoglobin beta-h0 chain [Mus musculus].
Modification Date	18-APR-2005
Weight	16,412 kDa

1.2 Half-life

N-terminal aa	Half-life mammals	Half-life yeast	Half-life E.Coli

Figure 14.16: *Comparative sequence statistics.*

Nucleotide sequence statistics are generated using the same dialog as used for protein sequence statistics. However, the output of Nucleotide sequence statistics is less extensive than that of the protein sequence statistics.

Note! The headings of the tables change depending on whether you calculate 'individual' or 'comparative' sequence statistics.

The output of comparative protein sequence statistics include:

- Sequence information:
  - Sequence type

  - Length
  - Organism
  - Name
  - Description
  - Modification Date
  - Weight
  - Isoelectric point
  - Aliphatic index

- Half-life
- Extinction coefficient
- Counts of Atoms
- Frequency of Atoms
- Count of hydrophobic and hydrophilic residues
- Frequencies of hydrophobic and hydrophilic residues
- Count of charged residues
- Frequencies of charged residues
- Amino acid distribution
- Histogram of amino acid distribution
- Annotation table
- Counts of di-peptides
- Frequency of di-peptides

The output of nucleotide sequence statistics include:

- General statistics:
  - Sequence type
  - Length
  - Organism
  - Name
  - Description
  - Modification Date
  - Weight (calculated as single-stranded DNA)
- Atomic composition
- Nucleotide distribution table

- Nucleotide distribution histogram
- Annotation table
- Counts of di-nucleotides
- Frequency of di-nucleotides

A short description of the different areas of the statistical output is given in section 14.4.1.

14.4.1 Bioinformatics explained: Protein statistics

Every protein holds specific and individual features which are unique to that particular protein. Features such as isoelectric point or amino acid composition can reveal important information of a novel protein. Many of the features described below are calculated in a simple way.

Molecular weight

The molecular weight is the mass of a protein or molecule. The molecular weight is simply calculated as the sum of the atomic mass of all the atoms in the molecule.

The weight of a protein is usually represented in Daltons (Da).

A calculation of the molecular weight of a protein does not usually include additional posttranslational modifications. For native and unknown proteins it tends to be difficult to assess whether posttranslational modifications such as glycosylations are present on the protein, making a calculation based solely on the amino acid sequence inaccurate. The molecular weight can be determined very accurately by mass-spectrometry in a laboratory.

Isoelectric point

The isoelectric point (pI) of a protein is the pH where the proteins has no net charge. The pI is calculated from the pKa values for 20 different amino acids. At a pH below the pI, the protein carries a positive charge, whereas if the pH is above pI the proteins carry a negative charge. In other words, pI is high for basic proteins and low for acidic proteins. This information can be used in the laboratory when running electrophoretic gels. Here the proteins can be separated, based on their isoelectric point.

Aliphatic index

The aliphatic index of a protein is a measure of the relative volume occupied by aliphatic side chain of the following amino acids: alanine, valine, leucine and isoleucine. An increase in the aliphatic index increases the thermostability of globular proteins. The index is calculated by the following formula.

$$Aliphaticindex = X(Ala) + a * X(Val) + b * X(Leu) + b * (X)Ile$$

X(Ala), *X(Val)*, *X(Ile)* and *X(Leu)* are the amino acid compositional fractions. The constants a and b are the relative volume of valine (a=2.9) and leucine/isoleucine (b=3.9) side chains compared to the side chain of alanine [Ikai, 1980].

Amino acid	**Mammalian**	**Yeast**	**E. coli**
Ala (A)	4.4 hour	>20 hours	>10 hours
Cys (C)	1.2 hours	>20 hours	>10 hours
Asp (D)	1.1 hours	3 min	>10 hours
Glu (E)	1 hour	30 min	>10 hours
Phe (F)	1.1 hours	3 min	2 min
Gly (G)	30 hours	>20 hours	>10 hours
His (H)	3.5 hours	10 min	>10 hours
Ile (I)	20 hours	30 min	>10 hours
Lys (K)	1.3 hours	3 min	2 min
Leu (L)	5.5 hours	3 min	2 min
Met (M)	30 hours	>20 hours	>10 hours
Asn (N)	1.4 hours	3 min	>10 hours
Pro (P)	>20 hours	>20 hours	?
Gln (Q)	0.8 hour	10 min	>10 hours
Arg (R)	1 hour	2 min	2 min
Ser (S)	1.9 hours	>20 hours	>10 hours
Thr (T)	7.2 hours	>20 hours	>10 hours
Val (V)	100 hours	>20 hours	>10 hours
Trp (W)	2.8 hours	3 min	2 min
Tyr (Y)	2.8 hours	10 min	2 min

Table 14.2: **Estimated half life.** Half life of proteins where the N-terminal residue is listed in the first column and the half-life in the subsequent columns for mammals, yeast and *E. coli*.

Estimated half-life

The half life of a protein is the time it takes for the protein pool of that particular protein to be reduced to the half. The half life of proteins is highly dependent on the presence of the N-terminal amino acid, thus overall protein stability [Bachmair et al., 1986, Gonda et al., 1989, Tobias et al., 1991]. The importance of the N-terminal residues is generally known as the 'N-end rule'. The N-end rule and consequently the N-terminal amino acid, simply determines the half-life of proteins. The estimated half-life of proteins have been investigated in mammals, yeast and *E. coli* (see Table 14.2). If leucine is found N-terminally in mammalian proteins the estimated half-life is 5.5 hours.

Extinction coefficient

This measure indicates how much light is absorbed by a protein at a particular wavelength. The extinction coefficient is measured by UV spectrophotometry, but can also be calculated. The amino acid composition is important when calculating the extinction coefficient. The extinction coefficient is calculated from the absorbance of cysteine, tyrosine and tryptophan using the following equation:

$$Ext(Protein) = count(Cystine)*Ext(Cystine)+count(Tyr)*Ext(Tyr)+count(Trp)*Ext(Trp)$$

where Ext is the extinction coefficient of amino acid in question. At 280nm the extinction coefficients are: Cys=120, Tyr=1280 and Trp=5690.

This equation is only valid under the following conditions:

- pH 6.5
- 6.0 M guanidium hydrochloride
- 0.02 M phosphate buffer

The extinction coefficient values of the three important amino acids at different wavelengths are found in [Gill and von Hippel, 1989].

Knowing the extinction coefficient, the absorbance (optical density) can be calculated using the following formula:

$$Absorbance(Protein) = \frac{Ext(Protein)}{Molecular\ weight}$$

Two values are reported. The first value is computed assuming that all cysteine residues appear as half cystines, meaning they form di-sulfide bridges to other cysteines. The second number assumes that no di-sulfide bonds are formed.

Atomic composition

Amino acids are indeed very simple compounds. All 20 amino acids consist of combinations of only five different atoms. The atoms which can be found in these simple structures are: Carbon, Nitrogen, Hydrogen, Sulfur, Oxygen. The atomic composition of a protein can for example be used to calculate the precise molecular weight of the entire protein.

Total number of negatively charged residues (Asp+Glu)

At neutral pH, the fraction of negatively charged residues provides information about the location of the protein. Intracellular proteins tend to have a higher fraction of negatively charged residues than extracellular proteins.

Total number of positively charged residues (Arg+Lys)

At neutral pH, nuclear proteins have a high relative percentage of positively charged amino acids. Nuclear proteins often bind to the negatively charged DNA, which may regulate gene expression or help to fold the DNA. Nuclear proteins often have a low percentage of aromatic residues [Andrade et al., 1998].

Amino acid distribution

Amino acids are the basic components of proteins. The amino acid distribution in a protein is simply the percentage of the different amino acids represented in a particular protein of interest. Amino acid composition is generally conserved through family-classes in different organisms which can be useful when studying a particular protein or enzymes across species borders. Another interesting observation is that amino acid composition variate slightly between proteins from different subcellular localizations. This fact has been used in several computational methods, used for prediction of subcellular localization.

Annotation table

This table provides an overview of all the different annotations associated with the sequence and their incidence.

Dipeptide distribution

This measure is simply a count, or frequency, of all the observed adjacent pairs of amino acids (dipeptides) found in the protein. It is only possible to report neighboring amino acids. Knowledge on dipeptide composition have previously been used for prediction of subcellular localization.

Creative Commons License

14.5 Join sequences

CLC Genomics Workbench can join several nucleotide or protein sequences into one sequence. This feature can for example be used to construct "supergenes" for phylogenetic inference by joining several disjoint genes into one. Note, that when sequences are joined, all their annotations are carried over to the new spliced sequence.

Two (or more) sequences can be joined by:

select sequences to join | **Toolbox in the Menu Bar** | **General Sequence Analyses** | **Join sequences** ()

or **select sequences to join** | **right-click any selected sequence** | **Toolbox** | **General Sequence Analyses** | **Join sequences** ()

This opens the dialog shown in figure 14.17.

If you have selected some sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences from the selected elements. Click **Next** opens the dialog shown in figure 14.18.

In step 2 you can change the order in which the sequences will be joined. Select a sequence and use the arrows to move the selected sequence up or down.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

The result is shown in figure 14.19.

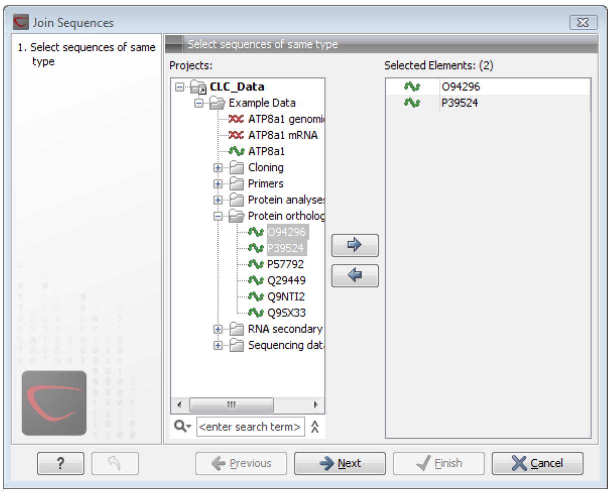


Figure 14.17: *Selecting two sequences to be joined.*

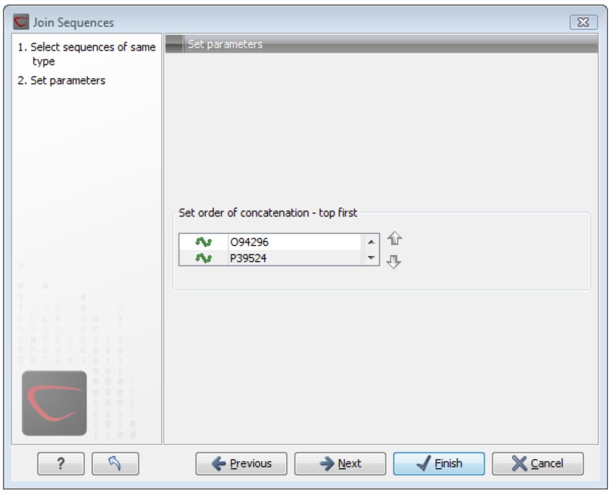


Figure 14.18: *Setting the order in which sequences are joined.*

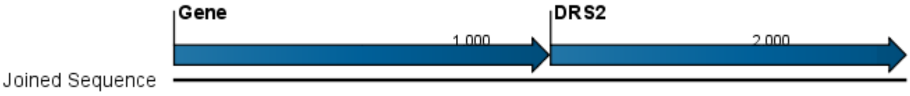


Figure 14.19: *The result of joining sequences is a new sequence containing the annotations of the joined sequences (they each had a HBB annotation).*

14.6 Pattern Discovery

With *CLC Genomics Workbench* you can perform pattern discovery on both DNA and protein sequences. Advanced hidden Markov models can help to identify unknown sequence patterns across single or even multiple sequences.

In order to search for unknown patterns:

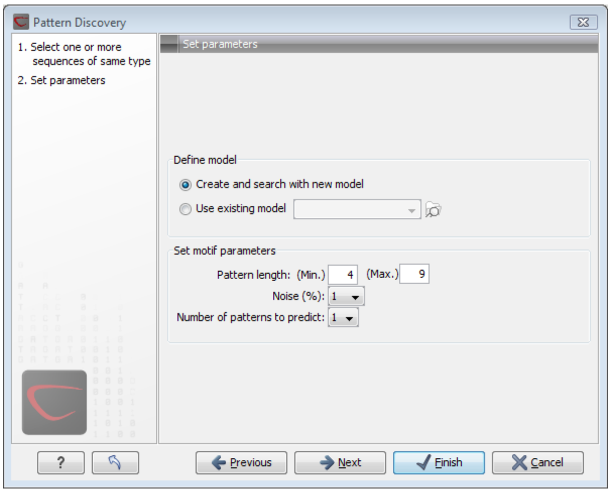
> **Select DNA or protein sequence(s)** | **Toolbox in the Menu Bar** | **General Sequence Analyses** ( ) | **Pattern Discovery** ( )

or **right-click DNA or protein sequence(s)** | **Toolbox** | **General Sequence Analyses** ( ) | **Pattern Discovery** ( )

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

You can perform the analysis on several DNA or several protein sequences at a time. If the analysis is performed on several sequences at a time the method will search for patterns which is common between all the sequences. Annotations will be added to all the sequences and a view is opened for each sequence.

Click **Next** to adjust parameters (see figure 14.20).



Define model

● Create and search with new model

○ Use existing model

Set motif parameters

Pattern length: (Min.) 4 (Max.) 9

Noise (%): 1

Number of patterns to predict: 1

Figure 14.20: *Setting parameters for the pattern discovery. See text for details.*

In order to search unknown sequences with an already existing model:

Select to use an already existing model which is seen in figure 14.20. Models are represented with the following icon in the navigation area ().

14.6.1 Pattern discovery search parameters

Various parameters can be set prior to the pattern discovery. The parameters are listed below and a screen shot of the parameter settings can be seen in figure 14.20.

- **Create and search with new model.** This will create a new HMM model based on the selected sequences. The found model will be opened after the run and presented in a table view. It can be saved and used later if desired.
- **Use existing model.** It is possible to use already created models to search for the same pattern in new sequences.
- **Minimum pattern length.** Here, the minimum length of patterns to search for, can be specified.
- **Maximum pattern length.** Here, the maximum length of patterns to search for, can be specified.
- **Noise (%).** Specify noise-level of the model. This parameter has influence on the level of degeneracy of patterns in the sequence(s). The noise parameter can be 1,2,5 or 10 percent.

- **Number of different kinds of patterns to predict.** Number of iterations the algorithm goes through. After the first iteration, we force predicted pattern-positions in the first run to be member of the background: In that way, the algorithm finds new patterns in the second iteration. Patterns marked 'Pattern1' have the highest confidence. The maximal iterations to go through is 3.

- **Include background distribution.** For protein sequences it is possible to include information on the background distribution of amino acids from a range of organisms.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will open a view showing the patterns found as annotations on the original sequence (see figure 14.21). If you have selected several sequences, a corresponding number of views will be opened.

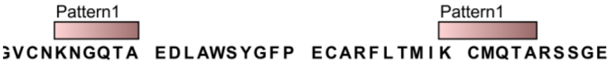


Figure 14.21: *Sequence view displaying two discovered patterns.*

14.6.2 Pattern search output

If the analysis is performed on several sequences at a time the method will search for patterns in the sequences and open a new view for each of the sequences, in which a pattern was discovered. Each novel pattern will be represented as an annotation of the type **Region**. More information on each found pattern is available through the tool-tip, including detailed information on the position of the pattern and quality scores.

It is also possible to get a tabular view of all found patterns in one combined table. Then each found pattern will be represented with various information on obtained scores, quality of the pattern and position in the sequence.

A table view of emission values of the actual used HMM model is presented in a table view. This model can be saved and used to search for a similar pattern in new or unknown sequences.

14.7 Motif Search

CLC Genomics Workbench offers advanced and versatile options to search for known motifs represented either by a simple sequence or a more advanced regular expression. These advanced search capabilities are available for use in both DNA and protein sequences.

There are two ways to access this functionality:

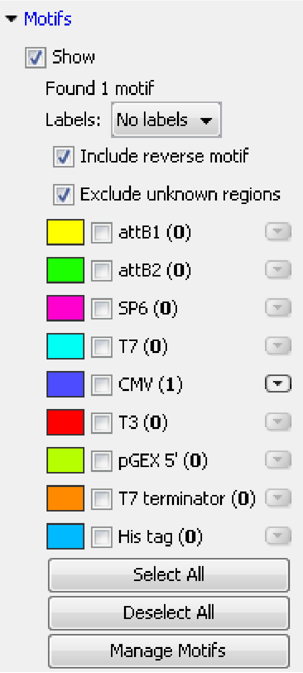
- When viewing sequences, it is possible to have motifs calculated and shown on the sequence in a similar way as restriction sites (see section 21.3.1). This approach is called *Dynamic motifs* and is an easy way to spot known sequence motifs when working with sequences for cloning etc.

- For more refined and systematic search for motifs can be performed through the **Toolbox**. This will generate a table and optionally add annotations to the sequences.

The two approaches are described below.

14.7.1 Dynamic motifs

In the **Side Panel** of sequence views, there is a group called **Motifs** (see figure 14.22).



▾ Motifs

☑ Show

Found 1 motif

Labels: No labels ▾

☑ Include reverse motif

☑ Exclude unknown regions

🟨 □ attB1 (0)

🟩 □ attB2 (0)

🟪 □ SP6 (0)

🟦 □ T7 (0)

🟦 □ CMV (1)

🟥 □ T3 (0)

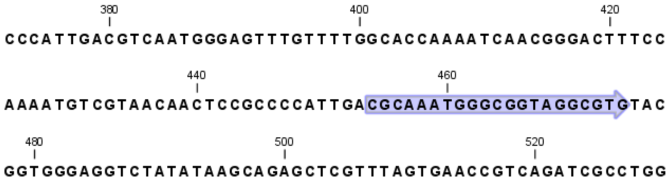
🟩 □ pGEX 5' (0)

🟧 □ T7 terminator (0)

🟦 □ His tag (0)

Figure 14.22: *Dynamic motifs in the Side Panel.*

The Workbench will look for the listed motifs in the sequence that is open and by clicking the check box next to the motif it will be shown in the view as illustrated in figure 14.23.

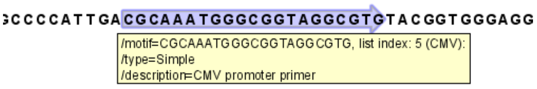


```
     380                 400                 420
CCCATTGACGTCAATGGGAGTTTGTTTTGGCACCAAAATCAACGGGACTTTCC

           440                 460
AAAATGTCGTAACAACTCCGCCCCATTGACGCAAATGGGCGGTAGGCGTGTAC

 480                 500                 520
GGTGGGAGGTCTATATAAGCAGAGCTCGTTTAGTGAACCGTCAGATCGCCTGG
```

Figure 14.23: *Showing dynamic motifs on the sequence.*

This case shows the CMV promoter primer sequence which is one of the pre-defined motifs in *CLC Genomics Workbench*. The motif is per default shown as a faded arrow with no text. The direction of the arrow indicates the strand of the motif.

Placing the mouse cursor on the arrow will display additional information about the motif as illustrated in figure 14.24.



```
GCCCCATTGACGCAAATGGGCGGTAGGCGTGTACGGTGGGAGG
```

/motif=CGCAAATGGGCGGTAGGCGTG, list index: 5 (CMV):
/type=Simple
/description=CMV promoter primer

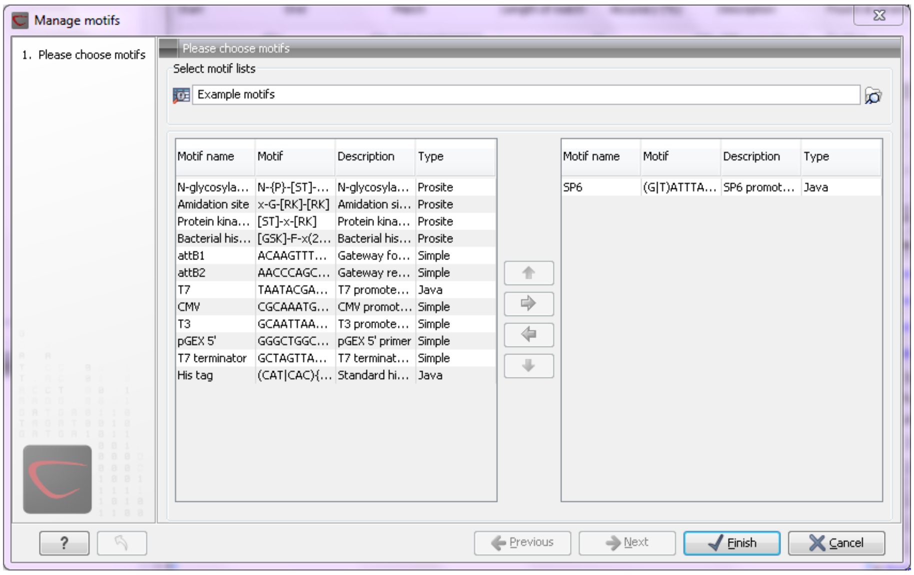
Figure 14.24: *Showing dynamic motifs on the sequence.*

To add **Labels** to the motif, select the **Flag** or **Stacked** option. They will put the name of the motif as a flag above the sequence. The stacked option will stack the labels when there is more than one motif so that all labels are shown.

Below the labels option there are two options for controlling the way the sequence should be searched for motifs:

- **Include reverse motifs**. This will also find motifs on the negative strand (only available for nucleotide sequences)
- **Exclude unknown regions**. The motif search handles ambiguous characters in the way that two residues are different if they do not have any residues in common. For example: For nucleotides, *N* matches any character and *R* matches *A,G*. For proteins, *X* matches any character and *Z* matches *E,Q*. Genome sequence often have large regions with unknown sequence. These regions are very often padded with N's. Ticking this checkbox will not display hits found in N-regions.

The list of motifs shown in figure 14.22 is a pre-defined list that is included with the *CLC Genomics Workbench*. You can define your own set of motifs to use instead. In order to do this, you first need to create a **Motif list** ( ) (see section 14.7.4) and then click the **Manage Motifs** button. This will bring up the dialog shown in figure 14.25.



Motif name	Motif	Description	Type
N-glycosyla...	N-{P}-[ST]-...	N-glycosyla...	Prosite
Amidation site	x-G-[RK]-[RK]	Amidation si...	Prosite
Protein kina...	[ST]-x-[RK]	Protein kina...	Prosite
Bacterial his...	[GSK]-F-x(2...	Bacterial his...	Prosite
attB1	ACAAGTTT...	Gateway fo...	Simple
attB2	AACCCAGC...	Gateway re...	Simple
T7	TAATACGA...	T7 promote...	Java
CMV	CGCAAATG...	CMV promot...	Simple
T3	GCAATTAA...	T3 promote...	Simple
pGEX 5'	GGGCTGGC...	pGEX 5' primer	Simple
T7 terminator	GCTAGTTA...	T7 terminat...	Simple
His tag	(CAT|CAC){...	Standard hi...	Java

Motif name	Motif	Description	Type
SP6	(G|T)ATTTA...	SP6 promot...	Java

Figure 14.25: *Managing the motifs to be shown.*

At the top, select a motif list by clicking the **Browse** ( ) button. When the motif list is selected, its motifs are listed in the panel in the left-hand side of the dialog. The right-hand side panel contains the motifs that will be listed in the **Side Panel** when you click **Finish**.

14.7.2 Motif search from the Toolbox

The dynamic motifs described in section 14.7.1 provide a quick way of routinely scanning a sequence for commonly used motifs, but in some cases a more systematic approach is needed. The motif search in the **Toolbox** provides an option to search for motifs with a user-specified similarity to the target sequence, and furthermore the motifs found can be displayed in an overview table. This is particularly useful when searching for motifs on many sequences.

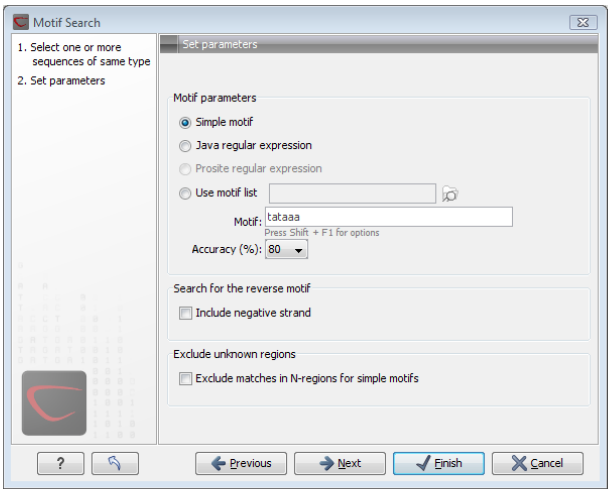
To start the Toolbox motif search:

Toolbox | **General Sequence Analyses** () | **Motif Search** ()

Use the arrows to add or remove sequences or sequence lists from the selected elements.

You can perform the analysis on several DNA or several protein sequences at a time. If the analysis is performed on several sequences at a time the method will search for patterns in the sequences and create an overview table of the motifs found in all sequences.

Click **Next** to adjust parameters (see figure 14.26).



Motif Search

1. Select one or more sequences of same type
2. Set parameters

Set parameters

Motif parameters

● Simple motif
○ Java regular expression
○ Prosite regular expression
○ Use motif list

Motif: tataaa

Press Shift + F1 for options

Accuracy (%): 80

Search for the reverse motif

□ Include negative strand

Exclude unknown regions

□ Exclude matches in N-regions for simple motifs

Figure 14.26: *Setting parameters for the motif search.*

The options for the motif search are:

- **Motif types.** Choose what kind of motif to be used:
  - Simple motif. Choosing this option means that you enter a simple motif, e.g. ATGATGNNATG.
  - Java regular expression. See section 14.7.3.
  - Prosite regular expression. For proteins, you can enter different protein patterns from the PROSITE database (protein patterns using regular expressions and describing specific amino acid sequences). The PROSITE database contains a great number of patterns and have been used to identify related proteins (see `http://www.expasy.org/cgi-bin/prosite-list.pl`).
  - Use motif list. Clicking the small button () will allow you to select a saved motif list (see section 14.7.4).
- **Motif.** If you choose to search with a simple motif, you should enter a literal string as your motif. Ambiguous amino acids and nucleotides are allowed. Example; ATGATGNNATG. If your motif type is Java regular expression, you should enter a regular expression according to the syntax rules described in section 14.7.3. Press **Shift + F1** key for options. For proteins, you can search with a Prosite regular expression and you should enter a protein pattern from the PROSITE database.

- **Accuracy.** If you search with a simple motif, you can adjust the accuracy of the motif to the match on the sequence. If you type in a simple motif and let the accuracy be 80%, the motif search algorithm runs through the input sequence and finds all subsequences of the same length as the simple motif such that the fraction of identity between the subsequence and the simple motif is at least 80%. A motif match is added to the sequence as an annotation with the exact fraction of identity between the subsequence and the simple motif. If you use a list of motifs, the accuracy applies only to the simple motifs in the list.

- **Search for reverse motif.** This enables searching on the negative strand on nucleotide sequences.

- **Exclude unknown regions.** Genome sequence often have large regions with unknown sequence. These regions are very often padded with N's. Ticking this checkbox will not display hits found in N-regions.Motif search handles ambiguous characters in the way that two residues are different if they do not have any residues in common. For example: For nucleotides, *N* matches any character and *R* matches *A,G*. For proteins, *X* matches any character and *Z* matches *E,Q*.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. There are two types of results that can be produced:

- **Add annotations**. This will add an annotation to the sequence when a motif is found (an example is shown in figure 14.27.

- **Create table**. This will create an overview table of all the motifs found for all the input sequences.

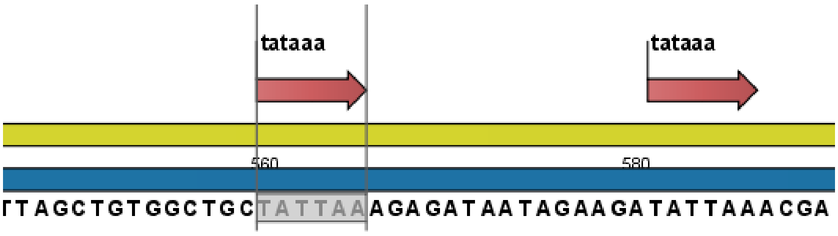


Figure 14.27: *Sequence view displaying the pattern found. The search string was 'tataaa'.*

14.7.3 Java regular expressions

A regular expressions is a string that describes or matches a set of strings, according to certain syntax rules. They are usually used to give a concise description of a set, without having to list all elements. The simplest form of a regular expression is a literal string. The syntax used for the regular expressions is the Java regular expression syntax (see `http://java.sun.com/docs/books/tutorial/essential/regex/index.html`). Below is listed some of the most important syntax rules which are also shown in the help pop-up when you press Shift + F1:

> *[A-Z]* will match the characters *A* through *Z* (Range). You can also put single characters between the brackets: The expression *[AGT]* matches the characters *A*, *G* or *T*.

[A-D[M-P]] will match the characters *A* through *D* and *M* through *P* (Union). You can also put single characters between the brackets: The expression *[AG[M-P]]* matches the characters *A*, *G* and *M* through *P*.

[A-M&&[H-P]] will match the characters between *A* and *M* lying between *H* and *P* (Intersection). You can also put single characters between the brackets. The expression *[A-M&&[HGTDA]]* matches the characters *A* through *M* which is *H*, *G*, *T*, *D* or *A*.

[^A-M] will match any character except those between *A* and *M* (Excluding). You can also put single characters between the brackets: The expression *[^AG]* matches any character except *A* and *G*.

[A-Z&&[^M-P]] will match any character *A* through *Z* except those between *M* and *P* (Subtraction). You can also put single characters between the brackets: The expression *[A-P&&[^CG]]* matches any character between *A* and *P* except *C* and *G*.

The symbol . matches any character.

X{n} will match a repetition of an element indicated by following that element with a numerical value or a numerical range between the curly brackets. For example, *ACG{2}* matches the string *ACGG* and *(ACG){2}* matches *ACGACG*.

X{n,m} will match a certain number of repetitions of an element indicated by following that element with two numerical values between the curly brackets. The first number is a lower limit on the number of repetitions and the second number is an upper limit on the number of repetitions. For example, *ACT{1,3}* matches *ACT, ACTT* and *ACTTT*.

X{n,} represents a repetition of an element at least *n* times. For example, *(AC){2,}* matches all strings *ACAC*, *ACACAC*, *ACACACAC*,...

The symbol ^ restricts the search to the beginning of your sequence. For example, if you search through a sequence with the regular expression *^AC*, the algorithm will find a match if *AC* occurs in the beginning of the sequence.

The symbol *$* restricts the search to the end of your sequence. For example, if you search through a sequence with the regular expression *GT$*, the algorithm will find a match if *GT* occurs in the end of the sequence.

Examples
The expression *[ACG][^AC]G{2}* matches all strings of length *4*, where the first character is *A,C* or *G* and the second is any character except *A,C* and the third and fourth character is *G*. The expression *G.[^A]$* matches all strings of length *3* in the end of your sequence, where the first character is *C*, the second any character and the third any character except *A*.

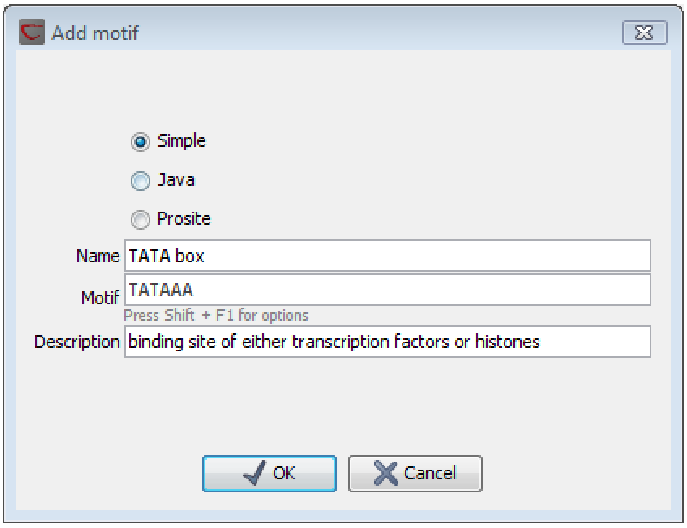
14.7.4 Create motif list

CLC Genomics Workbench offers advanced and versatile options to create lists of sequence patterns or known motifs represented either by a literal string or a regular expression.

A motif list is created from the Toolbox:

Toolbox | General Sequence Analyses | Create Motif List ()

This will open an empty list where you can add motifs by clicking the **Add** (✚) button at the bottom of the view. This will open a dialog shown in figure 14.28.



● Simple
○ Java
○ Prosite
Name: TATA box
Motif: TATAAA
Press Shift + F1 for options
Description: binding site of either transcription factors or histones

Figure 14.28: *Entering a new motif in the list.*

In this dialog, you can enter the following information:

- **Name**. The name of the motif. In the result of a motif search, this name will appear as the name of the annotation and in the result table.
- **Motif**. The actual motif. See section 14.7.2 for more information about the syntax of motifs.
- **Description**. You can enter a description of the motif. In the result of a motif search, the description will appear in the result table and added as a note to the annotation on the sequence (visible in the **Annotation table** (▤) or by placing the mouse cursor on the annotation).
- **Type**. You can enter three different types of motifs: Simple motifs, java regular expressions or PROSITE regular expression. Read more in section 14.7.2.

The motif list can contain a mix of different types of motifs. This is practical because some motifs can be described with the simple syntax, whereas others need the more advanced regular expression syntax.

Instead of manually adding motifs, you can **Import From Fasta File** (📂). This will show a dialog where you can select a fasta file on your computer and use this to create motifs. This will automatically take the name, description and sequence information from the fasta file, and put it into the motif list. The motif type will be "simple".

Besides adding new motifs, you can also edit and delete existing motifs in the list. To edit a motif, either double-click the motif in the list, or select and click the **Edit** (✎) button at the bottom of the view.

To delete a motif, select it and press the Delete key on the keyboard. Alternatively, click **Delete** (🗋) in the **Tool bar**.

Save the motif list in the **Navigation Area**, and you will be able to use for Motif Search (🔍) (see section 14.7).

Chapter 15

Nucleotide analyses

Contents

CLC Genomics Workbench offers different kinds of sequence analyses, which only apply to DNA and RNA.

15.1 Convert DNA to RNA

CLC Genomics Workbench lets you convert a DNA sequence into RNA, substituting the T residues (Thymine) for U residues (Urasil):

select a DNA sequence in the Navigation Area | Toolbox in the Menu Bar | Nucleotide Analyses () | Convert DNA to RNA ()

or **right-click a sequence in Navigation Area | Toolbox | Nucleotide Analyses () | Convert DNA to RNA ()**

This opens the dialog displayed in figure 15.1:

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Note! You can select multiple DNA sequences and sequence lists at a time. If the sequence list contains RNA sequences as well, they will not be converted.

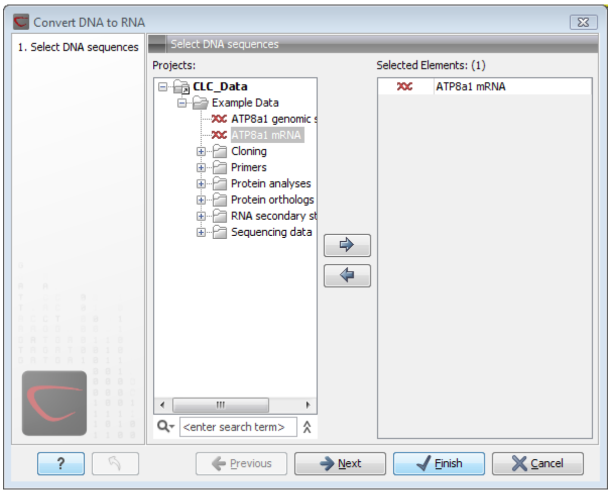


Figure 15.1: *Translating DNA to RNA.*

15.2 Convert RNA to DNA

CLC Genomics Workbench lets you convert an RNA sequence into DNA, substituting the U residues (Urasil) for T residues (Thymine):

> **select an RNA sequence in the Navigation Area | Toolbox in the Menu Bar | Nucleotide Analyses ( ) | Convert RNA to DNA ( )**

or **right-click a sequence in Navigation Area | Toolbox | Nucleotide Analyses ( ) | Convert RNA to DNA ( )**

This opens the dialog displayed in figure 15.2:

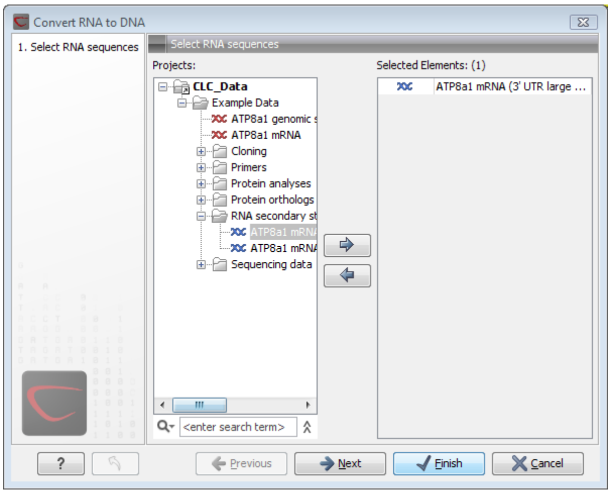


Figure 15.2: *Translating RNA to DNA.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

This will open a new view in the **View Area** displaying the new DNA sequence. The new sequence is not saved automatically. To save the sequence, drag it into the **Navigation Area** or press Ctrl

+ S (⌘ + S on Mac) to activate a save dialog.

Note! You can select multiple RNA sequences and sequence lists at a time. If the sequence list contains DNA sequences as well, they will not be converted.

15.3 Reverse complements of sequences

CLC Genomics Workbench is able to create the reverse complement of a nucleotide sequence. By doing that, a new sequence is created which also has all the annotations reversed since they now occupy the opposite strand of their previous location.

To quickly obtain the reverse complement of a sequence or part of a sequence, you may select a region on the negative strand and open it in a new view:

> **right-click a selection on the negative strand | Open selection in New View ()**

By doing that, the sequence will be reversed. This is only possible when the double stranded view option is enabled. It is possible to copy the selection and paste it in a word processing program or an e-mail. To obtain a reverse complement of an entire sequence:

> **select a sequence in the Navigation Area | Toolbox in the Menu Bar | Nucleotide Analyses () | Reverse Complement ()**

or **right-click a sequence in Navigation Area | Toolbox | Nucleotide Analyses () | Reverse Complement ()**

This opens the dialog displayed in figure 15.3:

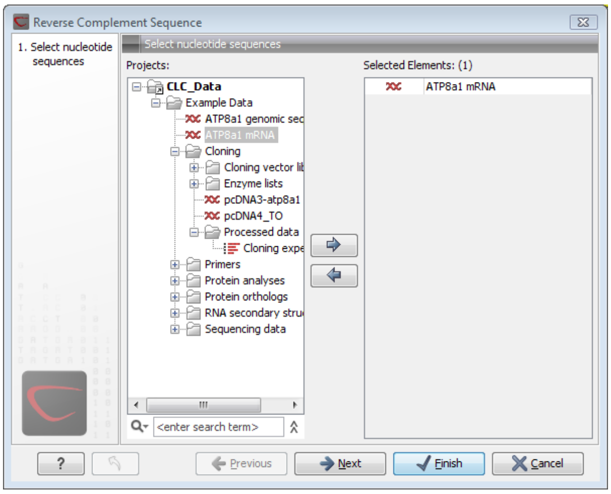


Figure 15.3: *Creating a reverse complement sequence.*

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

This will open a new view in the **View Area** displaying the reverse complement of the selected sequence. The new sequence is not saved automatically. To save the sequence, drag it into the **Navigation Area** or press Ctrl + S (⌘ + S on Mac) to activate a save dialog.

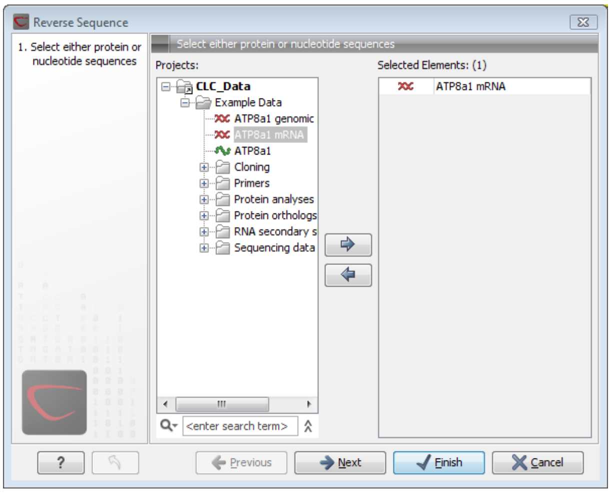
15.4 Reverse sequence

CLC Genomics Workbench is able to create the reverse of a nucleotide sequence. By doing that, a new sequence is created which also has all the annotations reversed since they now occupy the opposite strand of their previous location.

Note! This is not the same as a reverse complement. If you wish to create the reverse complement, please refer to section 15.3.

> **select a sequence in the Navigation Area | Toolbox in the Menu Bar | Nucleotide Analyses () | Reverse Sequence ()**

This opens the dialog displayed in figure 15.4:



1. Select either protein or nucleotide sequences

Selected Elements: (1)
ATP8a1 mRNA

Figure 15.4: *Reversing a sequence.*

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Note! This is not the same as a reverse complement. If you wish to create the reverse complement, please refer to section 15.3.

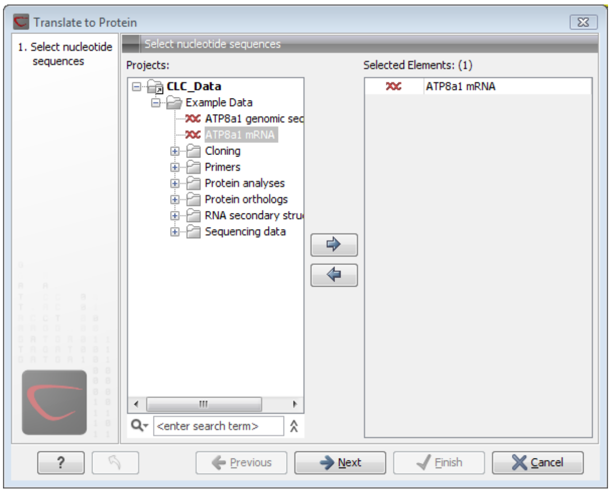
15.5 Translation of DNA or RNA to protein

In *CLC Genomics Workbench* you can translate a nucleotide sequence into a protein sequence using the **Toolbox** tools. Usually, you use the +1 reading frame which means that the translation starts from the first nucleotide. Stop codons result in an asterisk being inserted in the protein sequence at the corresponding position. It is possible to translate in any combination of the six reading frames in one analysis. To translate:

> **select a nucleotide sequence | Toolbox in the Menu Bar | Nucleotide Analyses () | Translate to Protein ()**

or **right-click a nucleotide sequence | Toolbox | Nucleotide Analyses () | Translate to Protein ()**

This opens the dialog displayed in figure 15.5:



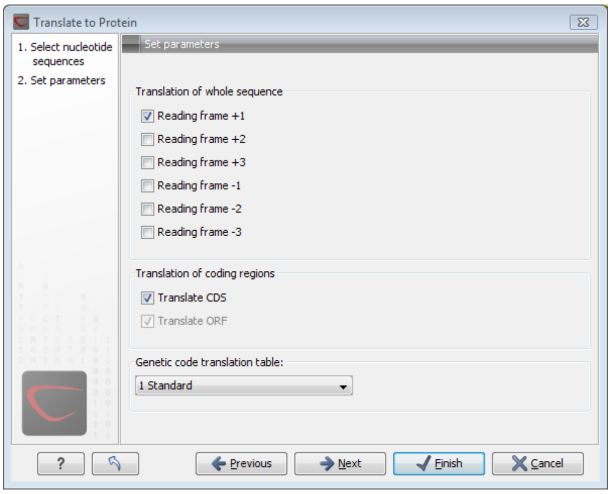
Selected Elements: (1)

ATP8a1 mRNA

Figure 15.5: *Choosing sequences for translation.*

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Clicking **Next** generates the dialog seen in figure 15.6:



Translation of whole sequence

☑ Reading frame +1
☐ Reading frame +2
☐ Reading frame +3
☐ Reading frame -1
☐ Reading frame -2
☐ Reading frame -3

Translation of coding regions

☑ Translate CDS
☑ Translate ORF

Genetic code translation table:

1 Standard

Figure 15.6: *Choosing +1 and +3 reading frames, and the standard translation table.*

Here you have the following options:

Reading frames If you wish to translate the whole sequence, you must specify the reading frame for the translation. If you select e.g. two reading frames, two protein sequences are generated.

Translate coding regions You can choose to translate regions marked by and CDS or ORF annotation. This will generate a protein sequence for each CDS or ORF annotation on the sequence.

Genetic code translation table Lets you specify the genetic code for the translation. The translation tables are occasionally updated from NCBI. The tables are not available in this printable version of the user manual. Instead, the tables are included in the **Help**-menu in the **Menu Bar** (in the appendix).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The newly created protein is shown, but is not saved automatically.

To save a protein sequence, drag it into the **Navigation Area** or press Ctrl + S (⌘ + S on Mac) to activate a save dialog.

15.5.1 Translate part of a nucleotide sequence

If you want to make separate translations of *all* the coding regions of a nucleotide sequence, you can check the option: "Translate CDS and ORF" in the translation dialog (see figure 15.6).

If you want to translate a *specific* coding region, which is annotated on the sequence, use the following procedure:

Open the nucleotide sequence | right-click the ORF or CDS annotation | Translate CDS/ORF () | choose a translation table | OK

If the annotation contains information about the translation, this information will be used, and you do not have to specify a translation table.

The CDS and ORF annotations are colored yellow as default.

15.6 Find open reading frames

CLC Genomics Workbench has a basic functionality for gene finding in the form of open reading frame (ORF) determination. The ORFs will be shown as annotations on the sequence. You have the option of choosing translation table, start codons, minimum length and other parameters for finding the ORFs. These parameters will be explained in this section.

To find open reading frames:

select a nucleotide sequence | Toolbox in the Menu Bar | Nucleotide Analyses () | Find Open Reading Frames ()

or **right-click a nucleotide sequence | Toolbox | Nucleotide Analyses () | Find Open Reading Frames ()**

This opens the dialog displayed in figure 15.7:

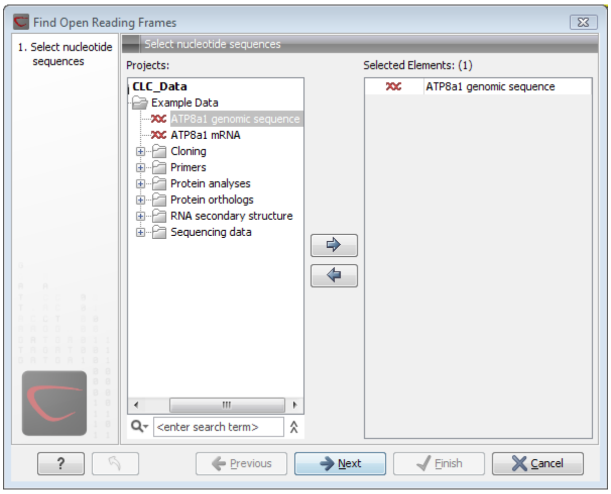
If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

If you want to adjust the parameters for finding open reading frames click **Next**.

15.6.1 Open reading frame parameters

This opens the dialog displayed in figure 15.8:

The adjustable parameters for the search are:



Projects:

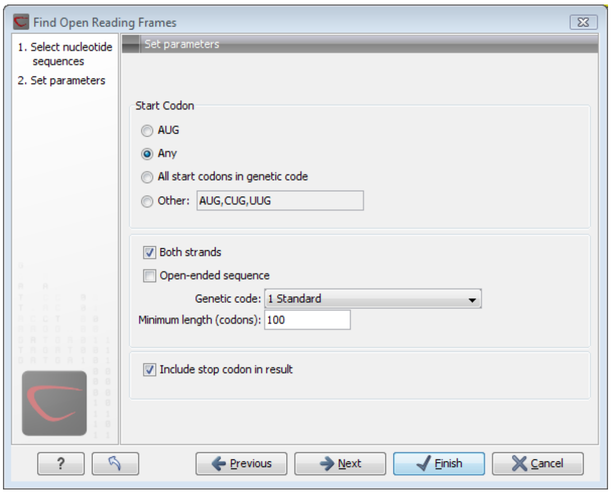
- CLC_Data
  - Example Data
    - ATP8a1 genomic sequence
    - ATP8a1 mRNA
    - Cloning
    - Primers
    - Protein analyses
    - Protein orthologs
    - RNA secondary structure
    - Sequencing data

<enter search term>

Selected Elements: (1)

ATP8a1 genomic sequence

Figure 15.7: *Create Reading Frame dialog.*



Start Codon

○ AUG

● Any

○ All start codons in genetic code

○ Other: AUG,CUG,UUG

☑ Both strands

□ Open-ended sequence

Genetic code: 1 Standard

Minimum length (codons): 100

☑ Include stop codon in result

Figure 15.8: *Create Reading Frame dialog.*

- **Start codon**:
  - **AUG**. Most commonly used start codon.
  - **Any**.
  - **All start codons in genetic code**.
  - **Other**. Here you can specify a number of start codons separated by commas.
- **Both strands**. Finds reading frames on both strands.
- **Open-ended Sequence**. Allows the ORF to start or end outside the sequence. If the sequence studied is a part of a larger sequence, it may be advantageous to allow the ORF to start or end outside the sequence.
- **Genetic code translation table**.
- **Include stop codon in result** The ORFs will be shown as annotations which can include the stop codon if this option is checked. The translation tables are occasionally updated from NCBI. The tables are not available in this printable version of the user manual. Instead, the tables are included in the **Help**-menu in the **Menu Bar** (in the appendix).

- **Minimum Length**. Specifies the minimum length for the ORFs to be found. The length is specified as number of codons.

Using open reading frames for gene finding is a fairly simple approach which is likely to predict genes which are not real. Setting a relatively high minimum length of the ORFs will reduce the number of false positive predictions, but at the same time short genes may be missed (see figure 15.9).

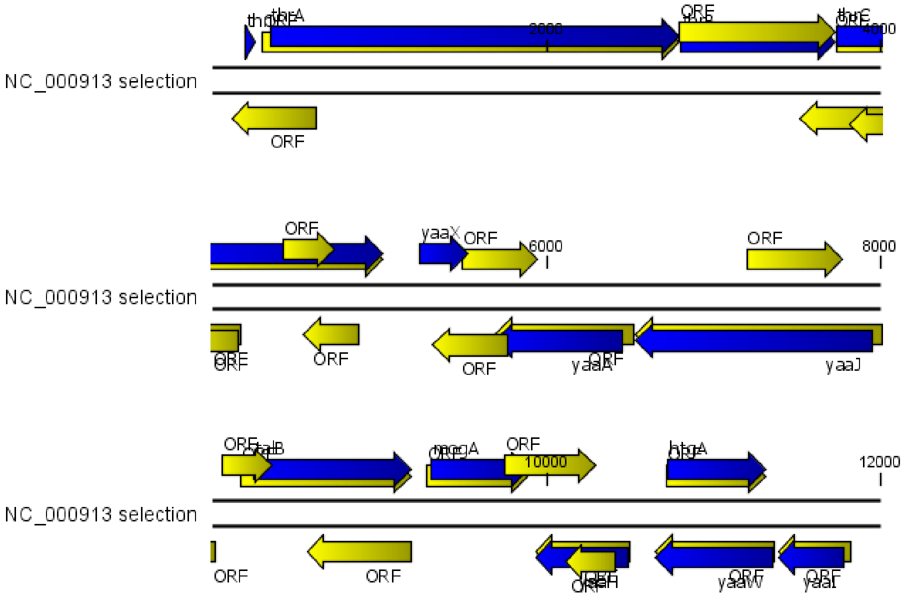


Figure 15.9: *The first 12,000 positions of the* E. coli *sequence NC_000913 downloaded from GenBank. The blue (dark) annotations are the genes while the yellow (brighter) annotations are the ORFs with a length of at least 100 amino acids. On the positive strand around position 11,000, a gene starts before the ORF. This is due to the use of the standard genetic code rather than the bacterial code. This particular gene starts with CTG, which is a start codon in bacteria. Two short genes are entirely missing, while a handful of open reading frames do not correspond to any of the annotated genes.*

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Finding open reading frames is often a good first step in annotating sequences such as cloning vectors or bacterial genomes. For eukaryotic genes, ORF determination may not always be very helpful since the intron/exon structure is not part of the algorithm.

Chapter 16

Protein analyses

Contents

CLC Genomics Workbench offers a number of analyses of proteins as described in this chapter.

16.1 Signal peptide prediction

Signal peptides target proteins to the extracellular environment either through direct plasmamembrane translocation in prokaryotes or is routed through the Endoplasmatic Reticulum in eukaryotic cells. The signal peptide is removed from the resulting mature protein during translocation across the membrane. For prediction of signal peptides, we query SignalP [Nielsen et al., 1997, Bendtsen et al., 2004b] located at `http://www.cbs.dtu.dk/services/SignalP/`. Thus an active internet connection is required to run the signal peptide prediction. Additional information on SignalP and Center for Biological Sequence analysis (CBS) can be found at `http://www.cbs.dtu.dk` and in the original research papers [Nielsen et al., 1997, Bendtsen et al., 2004b].

In order to predict potential signal peptides of proteins, the D-score from the SignalP output is used for discrimination of signal peptide versus non-signal peptide (see section 16.1.3). This score has been shown to be the most accurate [Klee and Ellis, 2005] in an evaluation study of signal peptide predictors.

In order to use SignalP, *you need to download the SignalP plug-in* using the plug-in manager, see section 1.7.1.

When the plug-in is downloaded and installed, you can use it to predict signal peptides:

Select a protein sequence | Toolbox in the Menu Bar | Protein Analyses () | Signal Peptide Prediction ()

or **right-click a protein sequence | Toolbox | Protein Analyses () | Signal Peptide Prediction ()**

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** to set parameters for the SignalP analysis.

16.1.1 Signal peptide prediction parameter settings

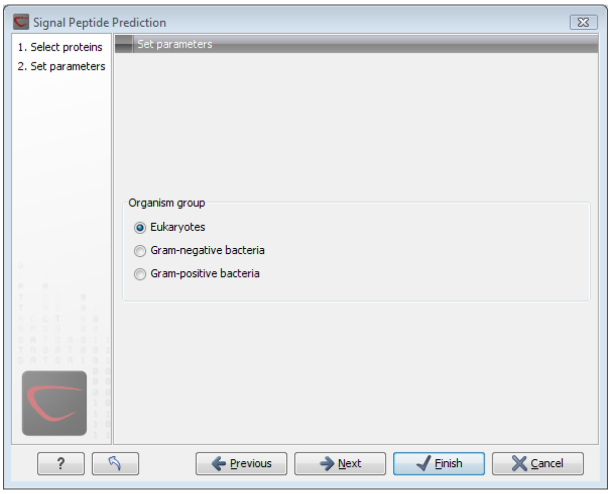
It is possible to set different options prior to running the analysis (see figure 16.1). An organism type should be selected. The default is eukaryote.

- Eukaryote (default)
- Gram-negative bacteria
- Gram-positive bacteria

You can perform the analysis on several protein sequences at a time. This will add annotations to all the sequences and open a view for each sequence if a signal peptide is found. If no signal peptide is found in the sequence a dialog box will be shown.

The predictions obtained can either be shown as annotations on the sequence, listed in a table or be shown as the detailed and full text output from the SignalP method. This can be used to interpret borderline predictions:

- Add annotations to sequence



Signal Peptide Prediction

1. Select proteins
2. Set parameters

Set parameters

Organism group

● Eukaryotes
○ Gram-negative bacteria
○ Gram-positive bacteria

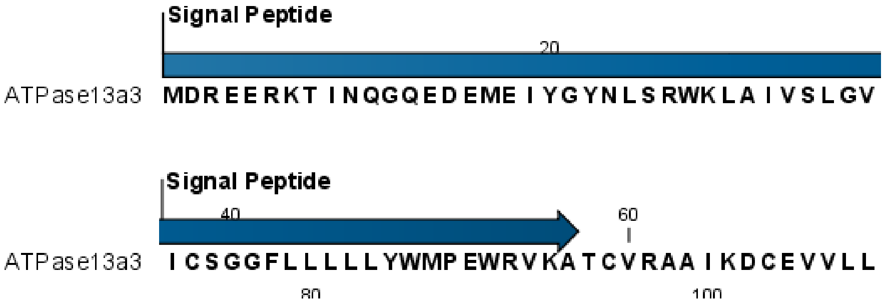
Figure 16.1: *Setting the parameters for signal peptide prediction.*

- Create table
- Text

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

16.1.2 Signal peptide prediction output

After running the prediction as described above, the protein sequence will show predicted signal peptide as annotations on the original sequence (see figure 16.2).



Signal Peptide
20
ATPase13a3
MDREERKTINQGQEDEMEIYGYNLSRWKLAIVSLGV
Signal Peptide
40
60
ATPase13a3
ICSGGFLLLLLYWMPEWRVKATCVRAAIKDCEVVLL
80
100

Figure 16.2: *N-terminal signal peptide shown as annotation on the sequence.*

Each annotation will carry a tooltip note saying that the corresponding annotation is predicted with SignalP version 3.0. Additional notes can be added through the **Edit annotation** ( ) right-click mouse menu. See section 10.3.2.

Undesired annotations can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 10.3.4.

16.1.3 Bioinformatics explained: Prediction of signal peptides

Why the interest in signal peptides?

The importance of signal peptides was shown in 1999 when Günter Blobel received the Nobel Prize in physiology or medicine for his discovery that "proteins have intrinsic signals that govern their transport and localization in the cell" [Blobel, 2000]. He pointed out the importance of defined peptide motifs for targeting proteins to their site of function.

Performing a query to PubMed[1] reveals that thousands of papers have been published, regarding signal peptides, secretion and subcellular localization, including knowledge of using signal peptides as vehicles for chimeric proteins for biomedical and pharmaceutical industry. Many papers describe statistical or machine learning methods for prediction of signal peptides and prediction of subcellular localization in general. After the first published method for signal peptide prediction [von Heijne, 1986], more and more methods have surfaced, although not all methods have been made available publicly.

Different types of signal peptides

Soon after Günter Blobel's initial discovery of signal peptides, more targeting signals were found. Most cell types and organisms employ several ways of targeting proteins to the extracellular environment or subcellular locations. Most of the proteins targeted for the extracellular space or subcellular locations carry specific sequence motifs (signal peptides) characterizing the type of secretion/targeting it undergoes.

Several new different signal peptides or targeting signals have been found during the later years, and papers often describe a small amino acid motif required for secretion of that particular protein. In most of the latter cases, the identified sequence motif is only found in this particular protein and as such cannot be described as a new group of signal peptides.

Describing the various types of signal peptides is beyond the scope of this text but several review papers on this topic can be found on PubMed. Targeting motifs can either be removed from, or retained in the mature protein after the protein has reached the correct and final destination. Some of the best characterized signal peptides are depicted in figure 16.3.

Numerous methods for prediction of protein targeting and signal peptides have been developed; some of them are mentioned and cited in the introduction of the SignalP research paper [Bendtsen et al., 2004b]. However, no prediction method will be able to cover all the different types of signal peptides. Most methods predicts classical signal peptides targeting to the general secretory pathway in bacteria or classical secretory pathway in eukaryotes. Furthermore, a few methods for prediction of non-classically secreted proteins have emerged [Bendtsen et al., 2004a, Bendtsen et al., 2005].

Prediction of signal peptides and subcellular localization

In the search for accurate prediction of signal peptides, many approaches have been investigated. Almost 20 years ago, the first method for prediction of classical signal peptides was published [von Heijne, 1986]. Nowadays, more sophisticated machine learning methods, such as neural networks, support vector machines, and hidden Markov models have arrived along with the increasing computational power and they all perform superior to the old weight matrix based methods [Menne et al., 2000]. Also, many other "classical" statistical approaches have been carried out, often in conjunction with machine learning methods. In the following sections, a wide range of different signal peptide and subcellular prediction methods will be described.

Most signal peptide prediction methods require the presence of the correct N-terminal end of

[1] http://www.ncbi.nlm.nih.gov/entrez/

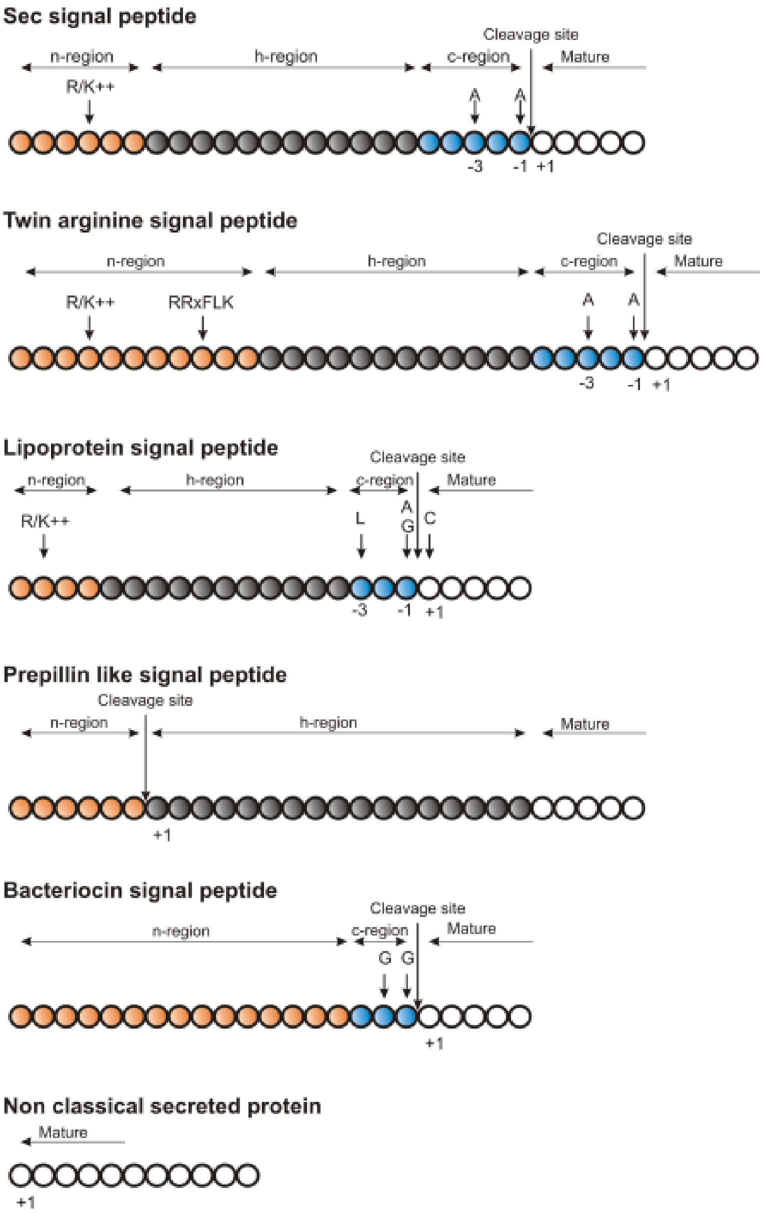


Figure 16.3: *Schematic representation of various signal peptides. Red color indicates n-region, gray color indicates h-region, cyan indicates c-region. All white circles are part of the mature protein.* $+1$ *indicates the first position of the mature protein. The length of the signal peptides is not drawn to scale.*

the preprotein for correct classification. As large scale genome sequencing projects sometimes assign the 5'-end of genes incorrectly, many proteins are annotated without the correct N-terminal [Reinhardt and Hubbard, 1998] leading to incorrect prediction of subcellular localization. These erroneous predictions can be ascribed directly to poor gene finding. Other methods for prediction of subcellular localization use information within the mature protein and therefore they are more robust to N-terminal truncation and gene finding errors.

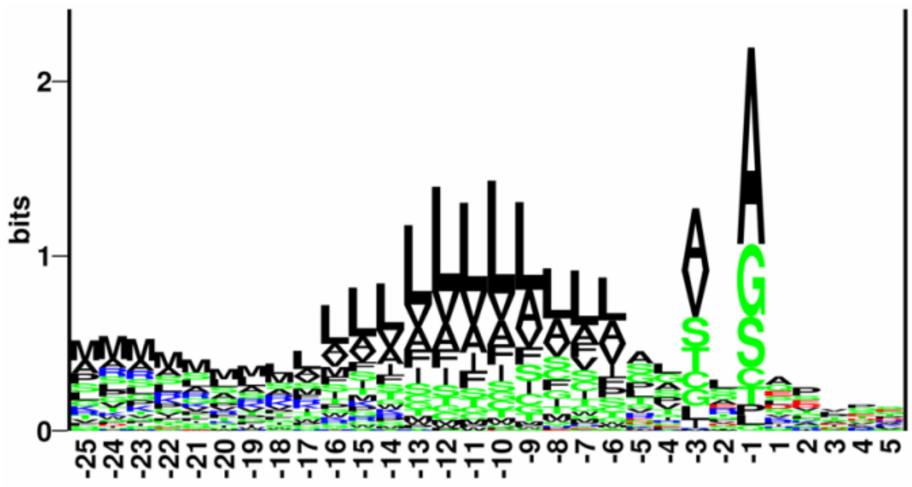


Figure 16.4: *Sequence logo of eukaryotic signal peptides, showing conservation of amino acids in bits [Schneider and Stephens, 1990]. Polar and hydrophobic residues are shown in green and black, respectively, while blue indicates positively charged residues and red negatively charged residues. The logo is based on an ungapped sequence alignment fixed at the -1 position of the signal peptides.*

The SignalP method

One of the most cited and best methods for prediction of classical signal peptides is the SignalP method [Nielsen et al., 1997, Bendtsen et al., 2004b]. In contrast to other methods, SignalP also predicts the actual cleavage site; thus the peptide which is cleaved off during translocation over the membrane. Recently, an independent research paper has rated SignalP version 3.0 to be the best standalone tool for signal peptide prediction. It was shown that the D-score which is reported by the SignalP method is the best measure for discriminating secretory from non-secretory proteins [Klee and Ellis, 2005].

SignalP is located at `http://www.cbs.dtu.dk/services/SignalP/`

What do the SignalP scores mean?

Many bioinformatics approaches or prediction tools do not give a yes/no answer. Often the user is facing an interpretation of the output, which can be either numerical or graphical. Why is that? In clear-cut examples there are no doubt; yes: this is a signal peptide! But, in borderline cases it is often convenient to have more information than just a yes/no answer. Here a graphical output can aid to interpret the correct answer. An example is shown in figure 16.5.

The graphical output from SignalP (neural network) comprises three different scores, *C*, *S* and *Y*. Two additional scores are reported in the SignalP3-NN output, namely the *S-mean* and the *D-score*, but these are only reported as numerical values.

For each organism class in SignalP; Eukaryote, Gram-negative and Gram-positive, two different neural networks are used, one for predicting the actual signal peptide and one for predicting the position of the signal peptidase I (SPase I) cleavage site. The *S-score* for the signal peptide prediction is reported for every single amino acid position in the submitted sequence, with high scores indicating that the corresponding amino acid is part of a signal peptide, and low scores indicating that the amino acid is part of a mature protein.
)

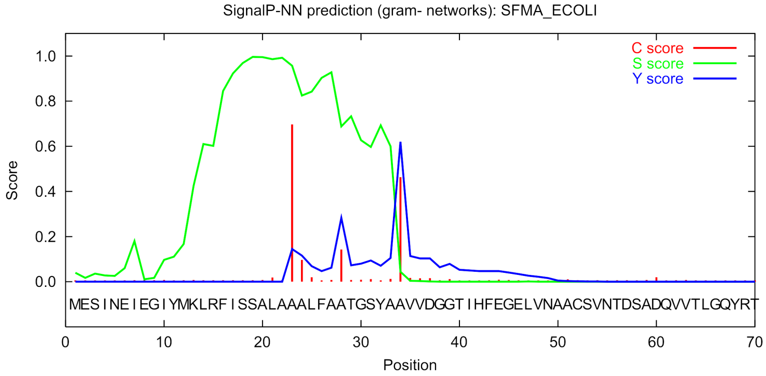


Figure 16.5: *Graphical output from the SignalP method of Swiss-Prot entry `SFMA_ECOLI`. Initially this seemed like a borderline prediction, but closer inspection of the sequence revealed an internal methionine at position 12, which could indicate a erroneously annotated start of the protein. Later this protein was re-annotated by Swiss-Prot to start at the M in position 12. See the text for description of the scores.*

The *C-score* is the "cleavage site" score. For each position in the submitted sequence, a C-score is reported, which should only be significantly high at the cleavage site. Confusion is often seen with the position numbering of the cleavage site. When a cleavage site position is referred to by a single number, the number indicates the first residue in the mature protein. This means that a reported cleavage site between amino acid 26-27 corresponds to the mature protein starting at (and include) position 27.

Y-max is a derivative of the C-score combined with the S-score resulting in a better cleavage site prediction than the raw C-score alone. This is due to the fact that multiple high-peaking C-scores can be found in one sequence, where only one is the true cleavage site. The cleavage site is assigned from the Y-score where the slope of the S-score is steep and a significant C-score is found.

The *S-mean* is the average of the S-score, ranging from the N-terminal amino acid to the amino acid assigned with the highest Y-max score, thus the S-mean score is calculated for the length of the predicted signal peptide. The S-mean score was in SignalP version 2.0 used as the criteria for discrimination of secretory and non-secretory proteins.

The *D-score* is introduced in SignalP version 3.0 and is a simple average of the S-mean and Y-max score. The score shows superior discrimination performance of secretory and non-secretory proteins to that of the S-mean score which was used in SignalP version 1 and 2.

For non-secretory proteins all the scores represented in the SignalP3-NN output should ideally be very low.

The hidden Markov model calculates the probability of whether the submitted sequence contains a signal peptide or not. The eukaryotic HMM model also reports the probability of a signal anchor, previously named uncleaved signal peptides. Furthermore, the cleavage site is assigned by a probability score together with scores for the n-region, h-region, and c-region of the signal peptide, if it is found.

Other useful resources

http://www.cbs.dtu.dk/services/SignalP

Pubmed entries for some of the original papers.

http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=pubmed&cmd=Retrieve&dopt=AbstractPlus&list_uids=9051728&query_hl=1&itool=pubmed_docsum

http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&list_uids=15223320&dopt=Citation

16.2 Protein charge

In *CLC Genomics Workbench* you can create a graph in the electric charge of a protein as a function of pH. This is particularly useful for finding the net charge of the protein at a given pH. This knowledge can be used e.g. in relation to isoelectric focusing on the first dimension of 2D-gel electrophoresis. The isoelectric point (pI) is found where the net charge of the protein is zero. The calculation of the protein charge does not include knowledge about any potential post-translational modifications the protein may have.

The pKa values reported in the literature may differ slightly, thus resulting in different looking graphs of the protein charge plot compared to other programs.

In order to calculate the protein charge:

Select a protein sequence | **Toolbox in the Menu Bar** | **Protein Analyses ( )** | **Create Protein Charge Plot ( )**

or **right-click a protein sequence** | **Toolbox** | **Protein Analyses ( )** | **Create Protein Charge Plot ( )**

This opens the dialog displayed in figure 16.6:

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

You can perform the analysis on several protein sequences at a time. This will result in one output graph showing protein charge graphs for the individual proteins.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

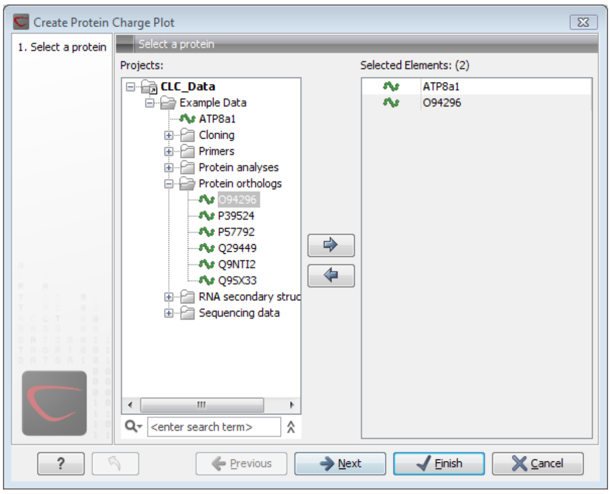


Figure 16.6: *Choosing protein sequences to calculate protein charge.*

16.2.1 Modifying the layout

Figure 16.7 shows the electrical charges for three proteins. In the **Side Panel** to the right, you can modify the layout of the graph.

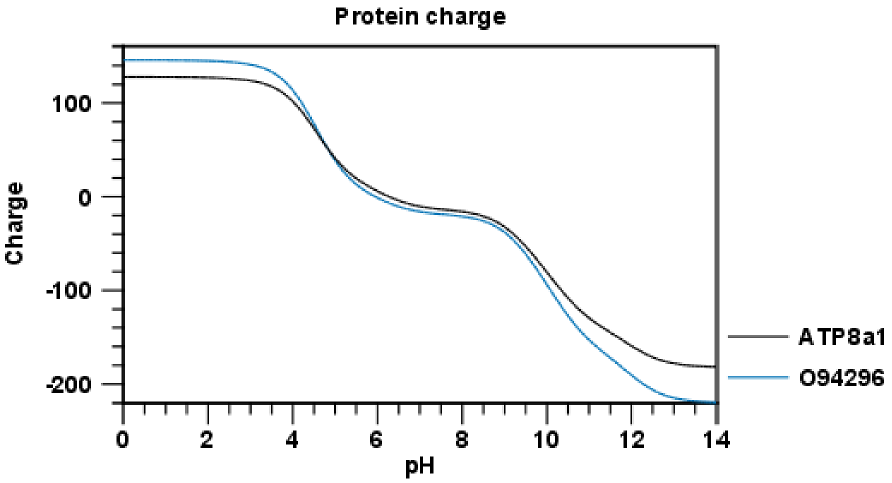


Figure 16.7: *View of the protein charge.*

See section B in the appendix for information about the graph view.

16.3 Transmembrane helix prediction

Many proteins are integral membrane proteins. Most membrane proteins have hydrophobic regions which span the hydrophobic core of the membrane bi-layer and hydrophilic regions located on the outside or the inside of the membrane. Many receptor proteins have several transmembrane helices spanning the cellular membrane.

For prediction of transmembrane helices, *CLC Genomics Workbench* uses TMHMM version 2.0 [Krogh et al., 2001] located at `http://www.cbs.dtu.dk/services/TMHMM/`, thus an active internet connection is required to run the transmembrane helix prediction. Additional information on THMHH and Center for Biological Sequence analysis (CBS) can be found at http://www.cbs.dtu.dk and in the original research paper [Krogh et al., 2001].

In order to use the transmembrane helix prediction, *you need to download the plug-in* using the plug-in manager (see section 1.7.1).

When the plug-in is downloaded and installed, you can use it to predict transmembrane helices:

> **Select a protein sequence | Toolbox in the Menu Bar | Protein Analyses ( ) | Transmembrane Helix Prediction ( )**

or **right-click a protein sequence | Toolbox | Protein Analyses ( ) | Transmembrane Helix Prediction ( )**

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

The predictions obtained can either be shown as annotations on the sequence, in a table or as the detailed and text output from the TMHMM method.

- Add annotations to sequence
- Create table
- Text

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

You can perform the analysis on several protein sequences at a time. This will add annotations to all the sequences and open a view for each sequence if a transmembrane helix is found. If a transmembrane helix is not found a dialog box will be presented.

After running the prediction as described above, the protein sequence will show predicted transmembrane helices as annotations on the original sequence (see figure 16.8). Moreover, annotations showing the topology will be shown. That is, which part the proteins is located on the inside or on the outside.

Each annotation will carry a tooltip note saying that the corresponding annotation is predicted with TMHMM version 2.0. Additional notes can be added through the **Edit annotation** ( ) right-click mouse menu. See section 10.3.2.

Undesired annotations can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 10.3.4.

16.4 Antigenicity

CLC Genomics Workbench can help to identify antigenic regions in protein sequences in different ways, using different algorithms. The algorithms provided in the Workbench, merely plot an index of antigenicity over the sequence.

Two different methods are available.

[Welling et al., 1985] Welling et al. used information on the relative occurrence of amino acids in antigenic regions to make a scale which is useful for prediction of antigenic regions. This method

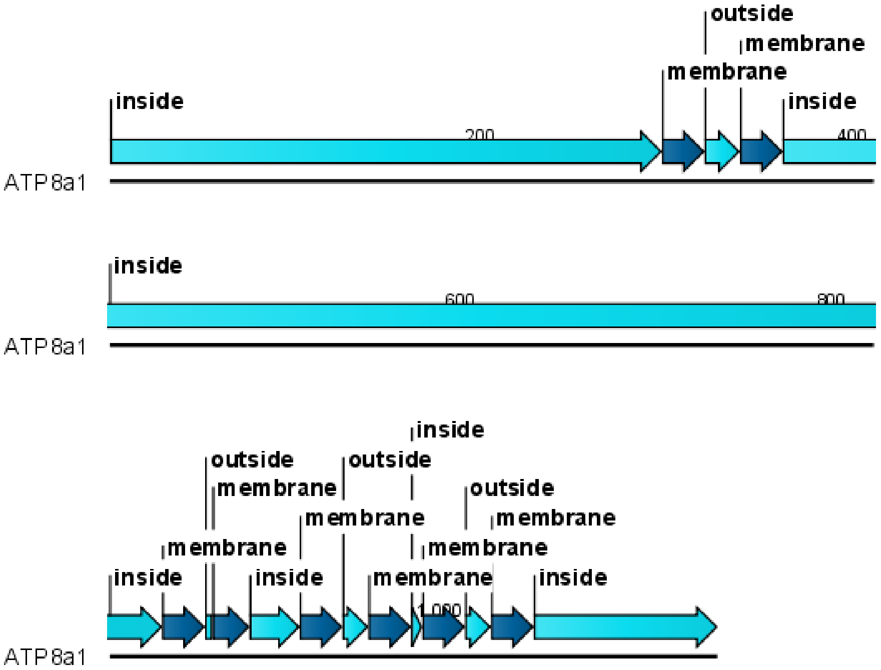


Figure 16.8: *Transmembrane segments shown as annotation on the sequence and the topology.*

is better than the Hopp-Woods scale of hydrophobicity which is also used to identify antigenic regions.

A semi-empirical method for prediction of antigenic regions has been developed [Kolaskar and Tongaonkar, 1990]. This method also includes information of surface accessibility and flexibility and at the time of publication the method was able to predict antigenic determinants with an accuracy of 75%.

Note! Similar results from the two method can not always be expected as the two methods are based on different training sets.

16.4.1 Plot of antigenicity

Displaying the antigenicity for a protein sequence in a plot is done in the following way:

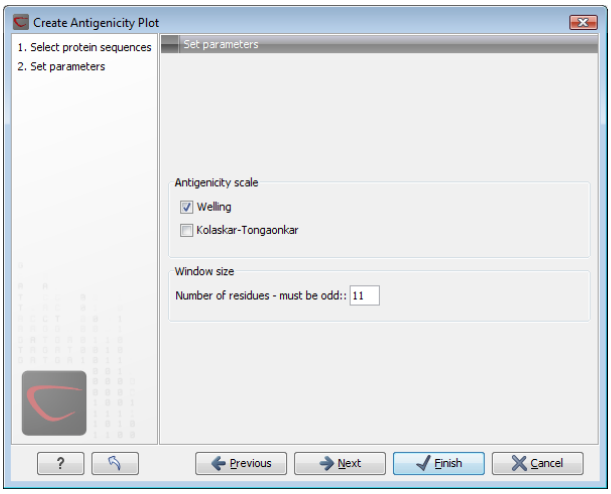
> **select a protein sequence in Navigation Area** | **Toolbox in the Menu Bar** | **Protein Analyses** ( ) | **Create Antigenicity Plot** ( )

This opens a dialog. The first step allows you to add or remove sequences. Clicking **Next** takes you through to **Step 2**, which is displayed in figure 16.9.

The **Window size** is the width of the window where, the antigenicity is calculated. The wider the window, the less volatile the graph. You can chose from a number of antigenicity scales. Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The result can be seen in figure 16.10.

See section B in the appendix for information about the graph view.

The level of antigenicity is calculated on the basis of the different scales. The different scales add different values to each type of amino acid. The antigenicity score is then calculated as the



Create Antigenicity Plot

1. Select protein sequences
2. Set parameters

Set parameters

Antigenicity scale

☑ Welling

□ Kolaskar-Tongaonkar

Window size

Number of residues - must be odd:: 11

Figure 16.9: *Step two in the Antigenicity Plot allows you to choose different antigenicity scales and the window size.*

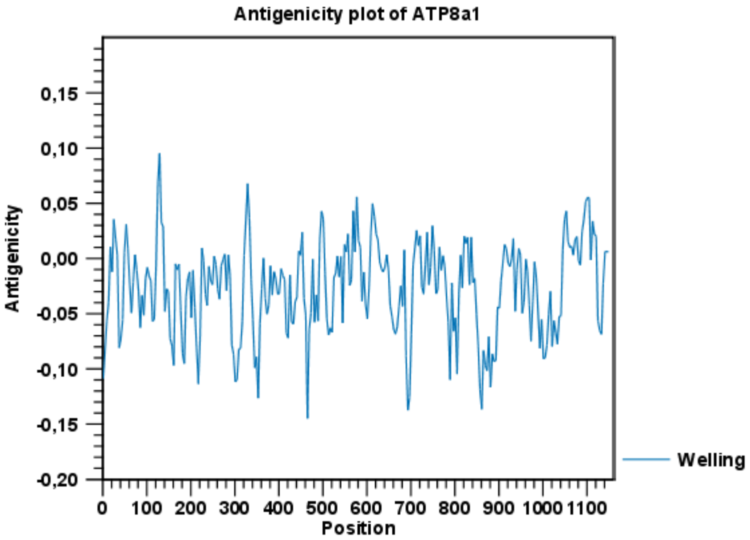


Figure 16.10: *The result of the antigenicity plot calculation and the associated Side Panel.*

sum of the values in a 'window', which is a particular range of the sequence. The window length can be set from 5 to 25 residues. The wider the window, the less fluctuations in the antigenicity scores.

16.4.2 Antigenicity graphs along sequence

Antigenicity graphs along the sequence can be displayed using the **Side Panel**. The functionality is similar to hydrophobicity (see section 16.5.2).

16.5 Hydrophobicity

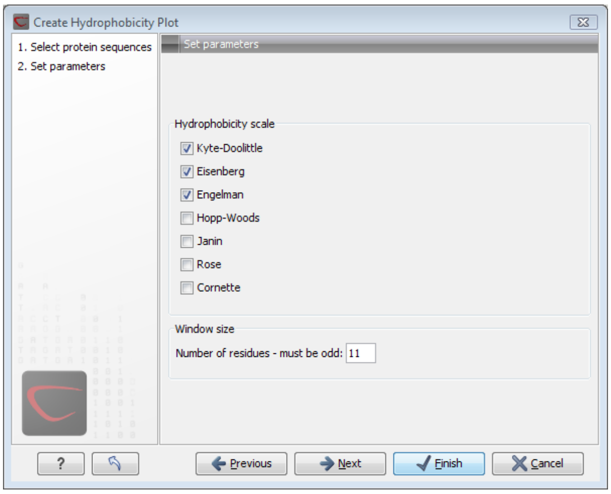
CLC Genomics Workbench can calculate the hydrophobicity of protein sequences in different ways, using different algorithms. (See section 16.5.3). Furthermore, hydrophobicity of sequences can be displayed as hydrophobicity plots and as graphs along sequences. In addition, *CLC Genomics Workbench* can calculate hydrophobicity for several sequences at the same time, and for alignments.

16.5.1 Hydrophobicity plot

Displaying the hydrophobicity for a protein sequence in a plot is done in the following way:

> **select a protein sequence in Navigation Area | Toolbox in the Menu Bar | Protein Analyses () | Create Hydrophobicity Plot ()**

This opens a dialog. The first step allows you to add or remove sequences. Clicking **Next** takes you through to **Step 2**, which is displayed in figure 16.11.



Create Hydrophobicity Plot

1. Select protein sequences
2. Set parameters

Set parameters

Hydrophobicity scale

- ☑ Kyte-Doolittle
- ☑ Eisenberg
- ☑ Engelman
- ☐ Hopp-Woods
- ☐ Janin
- ☐ Rose
- ☐ Cornette

Window size

Number of residues - must be odd: 11

Figure 16.11: *Step two in the Hydrophobicity Plot allows you to choose hydrophobicity scale and the window size.*

The **Window size** is the width of the window where the hydrophobicity is calculated. The wider the window, the less volatile the graph. You can chose from a number of hydrophobicity scales which are further explained in section 16.5.3 Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The result can be seen in figure 16.12.

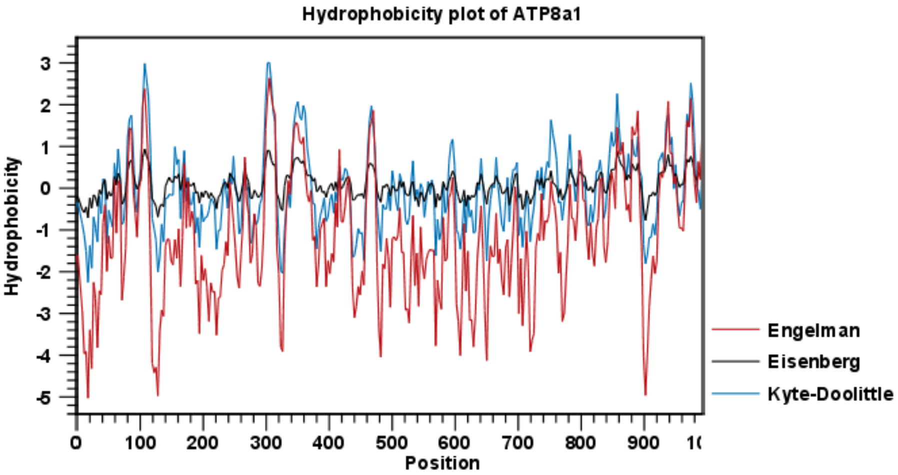


Figure 16.12: *The result of the hydrophobicity plot calculation and the associated Side Panel.*

See section B in the appendix for information about the graph view.

16.5.2 Hydrophobicity graphs along sequence

Hydrophobicity graphs along sequence can be displayed easily by activating the calculations from the **Side Panel** for a sequence.

> **right-click protein sequence in Navigation Area | Show | Sequence | open Protein info in Side Panel**

or **double-click protein sequence in Navigation Area | Show | Sequence | open Protein info in Side Panel**

These actions result in the view displayed in figure 16.13.

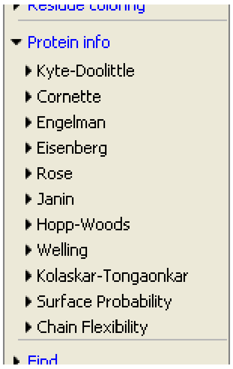


Figure 16.13: *The different available scales in Protein info in* **CLC Genomics Workbench***.*

The level of hydrophobicity is calculated on the basis of the different scales. The different scales add different values to each type of amino acid. The hydrophobicity score is then calculated as the sum of the values in a 'window', which is a particular range of the sequence. The window length can be set from 5 to 25 residues. The wider the window, the less fluctuations in the hydrophobicity scores. (For more about the theory behind hydrophobicity, see 16.5.3).

In the following we will focus on the different ways that *CLC Genomics Workbench* offers to display the hydrophobicity scores. We use Kyte-Doolittle to explain the display of the scores, but the different options are the same for all the scales. Initially there are three options for displaying the hydrophobicity scores. You can choose one, two or all three options by selecting the boxes. (See figure 16.14).

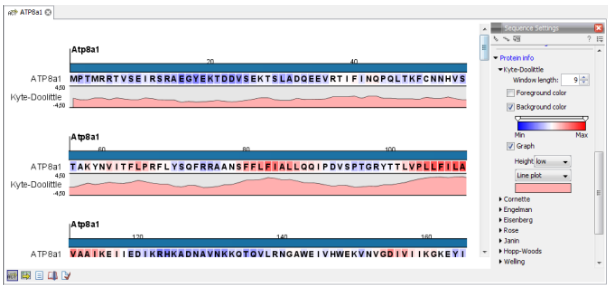


Figure 16.14: *The different ways of displaying the hydrophobicity scores, using the Kyte-Doolittle scale.*

Coloring the letters and their background. When choosing coloring of letters or coloring of their background, the color red is used to indicate high scores of hydrophobicity. A 'color-slider' allows you to amplify the scores, thereby emphasizing areas with high (or low, blue) levels of hydrophobicity. The color settings mentioned are default settings. By clicking the color bar just below the color slider you get the option of changing color settings.

Graphs along sequences. When selecting graphs, you choose to display the hydrophobicity scores underneath the sequence. This can be done either by a line-plot or bar-plot, or by coloring. The latter option offers you the same possibilities of amplifying the scores as applies for coloring of letters. The different ways to display the scores when choosing 'graphs' are displayed in figure 16.14. Notice that you can choose the height of the graphs underneath the sequence.

16.5.3 Bioinformatics explained: Protein hydrophobicity

Calculation of hydrophobicity is important to the identification of various protein features. This can be membrane spanning regions, antigenic sites, exposed loops or buried residues. Usually, these calculations are shown as a plot along the protein sequence, making it easy to identify the location of potential protein features.

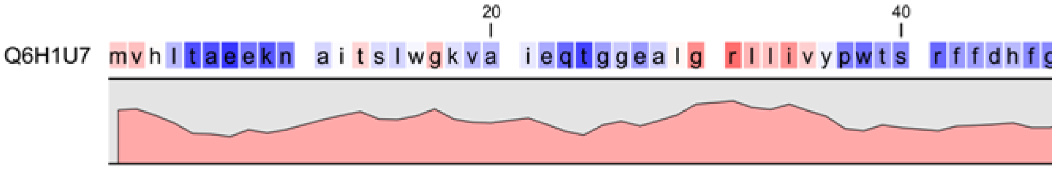


Figure 16.15: *Plot of hydrophobicity along the amino acid sequence. Hydrophobic regions on the sequence have higher numbers according to the graph below the sequence, furthermore hydrophobic regions are colored on the sequence. Red indicates regions with high hydrophobicity and blue indicates regions with low hydrophobicity.*

The hydrophobicity is calculated by sliding a fixed size window (of an odd number) over the protein sequence. At the central position of the window, the average hydrophobicity of the entire window is plotted (see figure 16.15).

Hydrophobicity scales

Several hydrophobicity scales have been published for various uses. Many of the commonly used hydrophobicity scales are described below.

Kyte-Doolittle scale. The Kyte-Doolittle scale is widely used for detecting hydrophobic regions in proteins. Regions with a positive value are hydrophobic. This scale can be used for identifying both surface-exposed regions as well as transmembrane regions, depending on the window size used. Short window sizes of 5-7 generally work well for predicting putative surface-exposed regions. Large window sizes of 19-21 are well suited for finding transmembrane domains if the values calculated are above 1.6 [Kyte and Doolittle, 1982]. These values should be used as a rule of thumb and deviations from the rule may occur.

Engelman scale. The Engelman hydrophobicity scale, also known as the GES-scale, is another scale which can be used for prediction of protein hydrophobicity [Engelman et al., 1986]. As the Kyte-Doolittle scale, this scale is useful for predicting transmembrane regions in proteins.

Eisenberg scale. The Eisenberg scale is a normalized consensus hydrophobicity scale which shares many features with the other hydrophobicity scales [Eisenberg et al., 1984].

Hopp-Woods scale. Hopp and Woods developed their hydrophobicity scale for identification of potentially antigenic sites in proteins. This scale is basically a hydrophilic index where apolar residues have been assigned negative values. Antigenic sites are likely to be predicted when using a window size of 7 [Hopp and Woods, 1983].

Cornette scale. Cornette *et al.* computed an optimal hydrophobicity scale based on 28 published scales [Cornette et al., 1987]. This optimized scale is also suitable for prediction of alpha-helices in proteins.

Rose scale. The hydrophobicity scale by Rose *et al.* is correlated to the average area of buried amino acids in globular proteins [Rose et al., 1985]. This results in a scale which is not showing the helices of a protein, but rather the surface accessibility.

Janin scale. This scale also provides information about the accessible and buried amino acid residues of globular proteins [Janin, 1979].

Welling scale. Welling et al. used information on the relative occurrence of amino acids in antigenic regions to make a scale which is useful for prediction of antigenic regions. This method is better than the Hopp-Woods scale of hydrophobicity which is also used to identify antigenic regions.

Kolaskar-Tongaonkar. A semi-empirical method for prediction of antigenic regions has been developed [Kolaskar and Tongaonkar, 1990]. This method also includes information of surface accessibility and flexibility and at the time of publication the method was able to predict antigenic determinants with an accuracy of 75%.

Surface Probability. Display of surface probability based on the algorithm by [Emini et al., 1985]. This algorithm has been used to identify antigenic determinants on the surface of proteins.

Chain Flexibility. isplay of backbone chain flexibility based on the algorithm by [Karplus and Schulz, 1985]. It is known that chain flexibility is an indication of a putative antigenic determinant.

Many more scales have been published throughout the last three decades. Even though more advanced methods have been developed for prediction of membrane spanning regions, the simple and very fast calculations are still highly used.

Other useful resources

AAindex: Amino acid index database
`http://www.genome.ad.jp/dbget/aaindex.html`

Creative Commons License

aa	aa	Kyte-Doolittle	Hopp-Woods	Cornette	Eisenberg	Rose	Janin	Engelman (GES)
A	Alanine	1.80	-0.50	0.20	0.62	0.74	0.30	1.60
C	Cysteine	2.50	-1.00	4.10	0.29	0.91	0.90	2.00
D	Aspartic acid	-3.50	3.00	-3.10	-0.90	0.62	-0.60	-9.20
E	Glutamic acid	-3.50	3.00	-1.80	-0.74	0.62	-0.70	-8.20
F	Phenylalanine	2.80	-2.50	4.40	1.19	0.88	0.50	3.70
G	Glycine	-0.40	0.00	0.00	0.48	0.72	0.30	1.00
H	Histidine	-3.20	-0.50	0.50	-0.40	0.78	-0.10	-3.00
I	Isoleucine	4.50	-1.80	4.80	1.38	0.88	0.70	3.10
K	Lysine	-3.90	3.00	-3.10	-1.50	0.52	-1.80	-8.80
L	Leucine	3.80	-1.80	5.70	1.06	0.85	0.50	2.80
M	Methionine	1.90	-1.30	4.20	0.64	0.85	0.40	3.40
N	Asparagine	-3.50	0.20	-0.50	-0.78	0.63	-0.50	-4.80
P	Proline	-1.60	0.00	-2.20	0.12	0.64	-0.30	-0.20
Q	Glutamine	-3.50	0.20	-2.80	-0.85	0.62	-0.70	-4.10
R	Arginine	-4.50	3.00	1.40	-2.53	0.64	-1.40	-12.3
S	Serine	-0.80	0.30	-0.50	-0.18	0.66	-0.10	0.60
T	Threonine	-0.70	-0.40	-1.90	-0.05	0.70	-0.20	1.20
V	Valine	4.20	-1.50	4.70	1.08	0.86	0.60	2.60
W	Tryptophan	-0.90	-3.40	1.00	0.81	0.85	0.30	1.90
Y	Tyrosine	-1.30	-2.30	3.20	0.26	0.76	-0.40	-0.70

Table 16.1: *Hydrophobicity scales. This table shows seven different hydrophobicity scales which are generally used for prediction of e.g. transmembrane regions and antigenicity.*

16.6 Pfam domain search

With *CLC Genomics Workbench* you can perform a search for Pfam domains on protein sequences. The Pfam database at http://pfam.sanger.ac.uk/ is a large collection of multiple sequence alignments that covers approximately 9318 protein domains and protein families [Bateman et al., 2004]. Based on the individual domain alignments, profile HMMs have been developed. These profile HMMs can be used to search for domains in unknown sequences.

Many proteins have a unique combination of domains which can be responsible, for instance, for the catalytic activities of enzymes. Pfam was initially developed to aid the annotation of the *C. elegans* genome. Annotating unknown sequences based on pairwise alignment methods by simply transferring annotation from a known protein to the unknown partner does not take domain organization into account [Galperin and Koonin, 1998]. An unknown protein may be annotated wrongly, for instance, as an enzyme if the pairwise alignment only finds a regulatory domain.

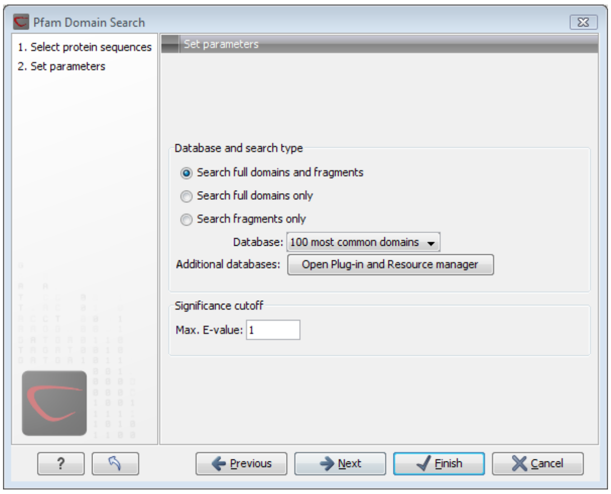
Using the Pfam search option in *CLC Genomics Workbench*, you can search for domains in sequence data which otherwise do not carry any annotation information. The Pfam search option adds all found domains onto the protein sequence which was used for the search. If domains of no relevance are found they can easily be removed as described in section 10.3.4. Setting a lower cutoff value will result in fewer domains.

In *CLC Genomics Workbench* we have implemented our own HMM algorithm for prediction of the Pfam domains. Thus, we do not use the original HMM implementation, HMMER http://hmmer.wustl.edu/ for domain prediction. We find the most probable state path/alignment through each profile HMM by the Viterbi algorithm and based on that we derive a new null model by averaging over the emission distributions of all *M* and *I* states that appear in the state path (*M* is a match state and *I* is an insert state). From that model we now arrive at an additive correction to the original bit-score, like it is done in the original HMMER algorithm.

In order to conduct the Pfam search:

> **Select a protein sequence** | **Toolbox in the Menu Bar** | **Protein Analyses ( )** | **Pfam Domain Search ( )**

or **right-click a protein sequence** | **Toolbox** | **Protein Analyses ( )** | **Pfam Domain Search ( )**



Database and search type

● Search full domains and fragments

○ Search full domains only

○ Search fragments only

Database: 100 most common domains

Additional databases: Open Plug-in and Resource manager

Significance cutoff

Max. E-value: 1

Figure 16.16: *Setting parameters for Pfam domain search.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

You can perform the analysis on several protein sequences at a time. This will add annotations to all the sequences and open a view for each sequence. Click **Next** to adjust parameters (see figure 16.16).

16.6.1 Pfam search parameters

- **Choose database and search type**
  When searching for Pfam domains it is possible to choose different databases and specify the search for full domains or fragments of domains. Only the 100 most frequent domains are included as default in *CLC Genomics Workbench*. Additional databases can be downloaded directly from CLC bio's web-site at http://www.clcbio.com/resources.
  - **Search full domains and fragments.** This option allows you to search both for full domain but also for partial domains. This could be the case if a domain extends beyond the ends of a sequence
  - **Search full domains only.** Selecting this option only allows searches for full domains.

  - **Search fragments only.** Only partial domains will be found.
  - **Database.** Only the 100 most frequent domains are included as default in *CLC Genomics Workbench*, but additional databases can be downloaded and installed as described in section 16.6.2.

- **Set significance cutoff.** The E-value (expectation value) is the number of hits that would be expected to have a score equal to or better than this value, by chance alone. This means that a good E-value which gives a confident prediction is much less than 1. E-values around 1 is what is expected by chance. Thus, the lower the E-value, the more specific the search for domains will be. Only positive numbers are allowed.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will open a view showing the found domains as annotations on the original sequence (see figure 16.17). If you have selected several sequences, a corresponding number of views will be opened.

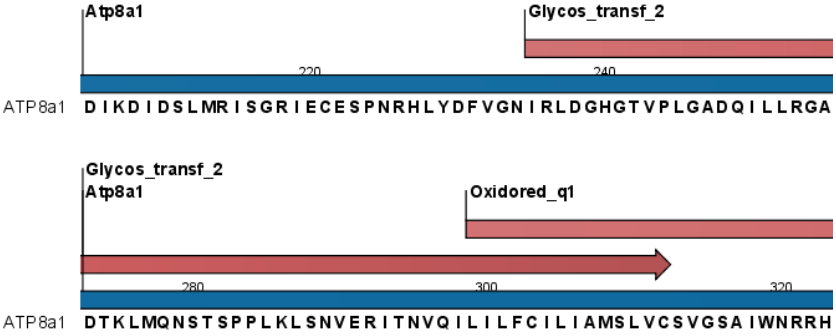

Atp8a1
Glycos_transf_2
220
240
ATP8a1
DIKDIDSLMRISGRIECESPNRHLYDFVGNIRLDGHGTVPLGADQILLRGA
Glycos_transf_2
Atp8a1
Oxidored_q1
280
300
320
ATP8a1
DTKLMQNSTSPPLKLSNVERITNVQILILFCILIAMSLVCSVGSAIWNRRH

Figure 16.17: *Domains annotations based on Pfam.*

Each found domain will be represented as an annotation of the type **Region**. More information on each found domain is available through the tooltip, including detailed information on the identity score which is the basis for the prediction.

For a more detailed description of the provided scores through the tooltip look at `http://www.sanger.ac.uk/Software/Pfam/help/scores.shtml`.

16.6.2 Download and installation of additional Pfam databases

Additional databases can be downloaded as a resource using the **Plug-in manager** ( ) (see section 1.7.4).

If you are not able to download directly from the Plug-in manager,

16.7 Secondary structure prediction

An important issue when trying to understand protein function is to know the actual structure of the protein. Many questions that are raised by molecular biologists are directly targeted at protein structure. The alpha-helix forms a coiled rodlike structure whereas a beta-sheet show an extended sheet-like structure. Some proteins are almost devoid of alpha-helices such as chymotrypsin (PDB_ID: 1AB9) whereas others like myoglobin (PDB_ID: 101M) have a very high content of alpha-helices.

With *CLC Genomics Workbench* one can predict the secondary structure of proteins very fast. Predicted elements are alpha-helix, beta-sheet (same as beta-strand) and other regions.

Based on extracted protein sequences from the protein databank (http://www.rcsb.org/pdb/) a hidden Makov model (HMM) was trained and evaluated for performance. Machine learning methods have shown superior when it comes to prediction of secondary structure of proteins [Rost, 2001]. By far the most common structures are Alpha-helices and beta-sheets which can be predicted, and predicted structures are automatically added to the query as annotation which later can be edited.

In order to predict the secondary structure of proteins:

Select a protein sequence | Toolbox in the Menu Bar | Protein Analyses () | Predict secondary structure ()

or **right-click a protein sequence | Toolbox | Protein Analyses () | Predict secondary structure ()**

This opens the dialog displayed in figure 16.18:

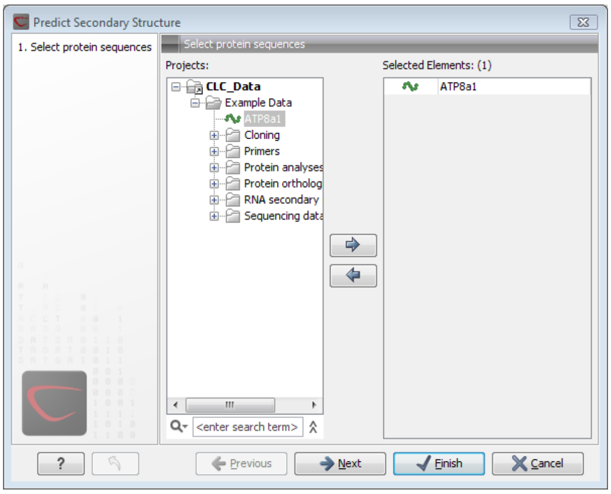


Figure 16.18: *Choosing one or more protein sequences for secondary structure prediction.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

You can perform the analysis on several protein sequences at a time. This will add annotations to all the sequences and open a view for each sequence.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

After running the prediction as described above, the protein sequence will show predicted alpha-helices and beta-sheets as annotations on the original sequence (see figure 16.19).

Each annotation will carry a tooltip note saying that the corresponding annotation is predicted with *CLC Genomics Workbench*. Additional notes can be added through the **Edit Annotation ()** right-click mouse menu. See section 10.3.2.

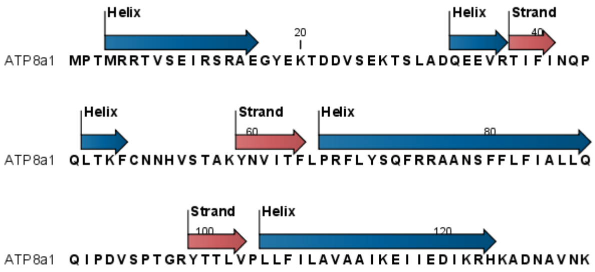

Helix
Helix
Strand
20
40
ATP8a1 MPTMRRTVSEIRSRAEGYEKTDDVSEKTSLADQEEVRTIFINQP
Helix
Strand
Helix
60
80
ATP8a1 QLTKFCNNHVSTAKYNVITFLPRFLYSQFRRAANSFFLFIALLQ
Strand
Helix
100
120
ATP8a1 QIPDVSPTGRYTTLVPLLFILAVAAIKEIIEDIKRHKADNAVNK

Figure 16.19: *Alpha-helices and beta-strands shown as annotations on the sequence.*

Undesired alpha-helices or beta-sheets can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 10.3.4.

16.8 Protein report

CLC Genomics Workbench is able to produce protein reports, that allow you to easily generate different kinds of information regarding a protein.

Actually a protein report is a collection of some of the protein analyses which are described elsewhere in this manual.

To create a protein report do the following:

> **Right-click protein in Navigation Area | Toolbox | Protein Analyses ( ) | Create Protein Report ( )**

This opens dialog **Step 1**, where you can choose which proteins to create a report for. When the correct one is chosen, click **Next**.

In dialog **Step 2** you can choose which analyses you want to include in the report. The following list shows which analyses are available and explains where to find more details.

- **Sequence statistics.** See section 14.4 for more about this topic.
- **Plot of charge as function of pH.** See section 16.2 for more about this topic.
- **Plot of hydrophobicity.** See section 16.5 for more about this topic.
- **Plot of local complexity.** See section 14.3 for more about this topic.
- **Dot plot against self.** See section 14.2 for more about this topic.
- **Secondary structure prediction.** See section 16.7 for more about this topic.
- **Pfam domain search.** See section 16.6 for more about this topic.
- **Local BLAST.** See section 12.2 for more about this topic.
- **NCBI BLAST.** See section 12.1 for more about this topic.

When you have selected the relevant analyses, click **Next**. **Step 3** to **Step 7** (if you select all the analyses in **Step 2**) are adjustments of parameters for the different analyses. The parameters are mentioned briefly in relation to the following steps, and you can turn to the relevant chapters or sections (mentioned above) to learn more about the significance of the parameters.

In **Step 3** you can adjust parameters for sequence statistics:

- **Individual Statistics Layout. Comparative** is disabled because reports are generated for one protein at a time.
- **Include Background Distribution of Amino Acids.** Includes distributions from different organisms. Background distributions are calculated from UniProt `www.uniprot.org` version 6.0, dated September 13 2005.

In **Step 4** you can adjust parameters for hydrophobicity plots:

- **Window size.** Width of window on sequence (odd number).
- **Hydrophobicity scales.** Lets you choose between different scales.

In **Step 5** you can adjust a parameter for complexity plots:

- **Window size.** Width of window on sequence (must be odd).

In **Step 6** you can adjust parameters for dot plots:

- **Score model.** Different scoring matrices.
- **Window size.** Width of window on sequence.

In **Step 7** you can adjust parameters for BLAST search:

- **Program.** Lets you choose between different BLAST programs.
- **Database.** Lets you limit your search to a particular database.

16.8.1 Protein report output

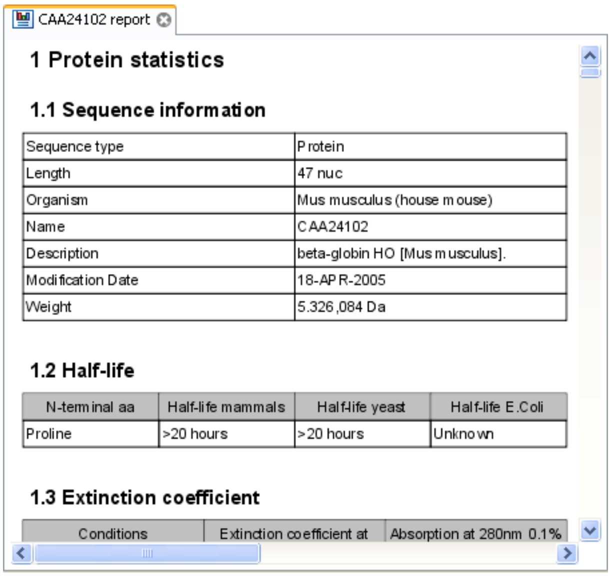
An example of Protein report can be seen in figure 16.20.

By double clicking a graph in the output, this graph is shown in a different view (*CLC Genomics Workbench* generates another tab). The report output and the new graph views can be saved by dragging the tab into the **Navigation Area**.

The content of the tables in the report can be copy/pasted out of the program and e.g. into Microsoft Excel. To do so:

> **Select content of table** | Right-click the selection | Copy

You can also **Export** () the report in Excel format.



1 Protein statistics

1.1 Sequence information

Sequence type	Protein
Length	47 nuc
Organism	Mus musculus (house mouse)
Name	CAA24102
Description	beta-globin HO [Mus musculus].
Modification Date	18-APR-2005
Weight	5.326,084 Da

1.2 Half-life

N-terminal aa	Half-life mammals	Half-life yeast	Half-life E.Coli
Proline	>20 hours	>20 hours	Unknown

1.3 Extinction coefficient

Conditions	Extinction coefficient at	Absorption at 280nm 0.1%

Figure 16.20: *A protein report. There is a Table of Contents in the Side Panel that makes it easy to browse the report.*

16.9 Reverse translation from protein into DNA

A protein sequence can be back-translated into DNA using *CLC Genomics Workbench*. Due to degeneracy of the genetic code every amino acid could translate into several different codons (only 20 amino acids but 64 different codons). Thus, the program offers a number of choices for determining which codons should be used. These choices are explained in this section.

In order to make a reverse translation:

> **Select a protein sequence | Toolbox in the Menu Bar | Protein Analyses ( ) | Reverse Translate ( )**

or **right-click a protein sequence | Toolbox | Protein Analyses ( ) | Reverse translate ( )**

This opens the dialog displayed in figure 16.21:

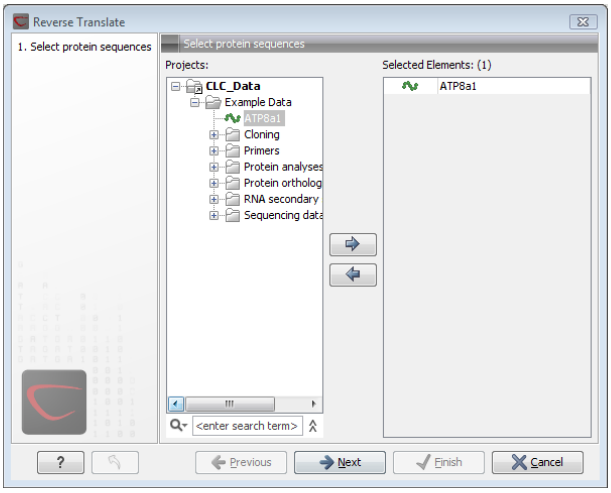


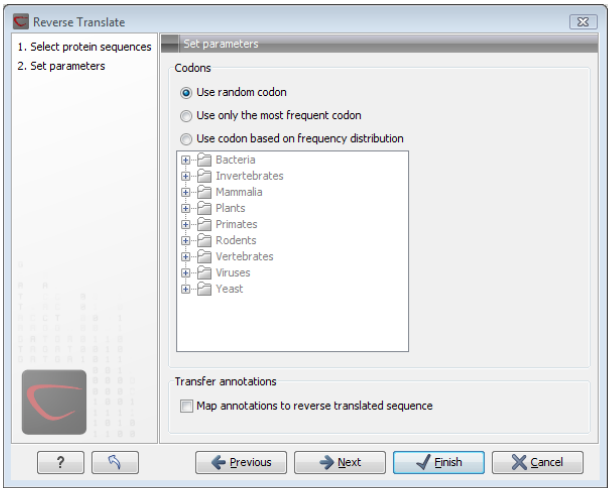
Figure 16.21: *Choosing a protein sequence for reverse translation.*

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements. You can translate several protein sequences at a time.

Click **Next** to adjust the parameters for the translation.

16.9.1 Reverse translation parameters

Figure 16.22 shows the choices for making the translation.



Reverse Translate

1. Select protein sequences
2. Set parameters

Set parameters

Codons

● Use random codon

○ Use only the most frequent codon

○ Use codon based on frequency distribution

Transfer annotations

□ Map annotations to reverse translated sequence

Figure 16.22: *Choosing parameters for the reverse translation.*

- **Use random codon.** This will randomly back-translate an amino acid to a codon without using the translation tables. Every time you perform the analysis you will get a different result.
- **Use only the most frequent codon.** On the basis of the selected translation table, this parameter/option will assign the codon that occurs most often. When choosing this option, the results of performing several reverse translations will always be the same, contrary to the other two options.
- **Use codon based on frequency distribution.** This option is a mix of the other two options. The selected translation table is used to attach weights to each codon based on its frequency. The codons are assigned randomly with a probability given by the weights. A more frequent codon has a higher probability of being selected. Every time you perform the analysis, you will get a different result. This option yields a result that is closer to the translation behavior of the organism (assuming you choose an appropriate codon frequency table).
- **Map annotations to reverse translated sequence.** If this checkbox is checked, then all annotations on the protein sequence will be mapped to the resulting DNA sequence. In the tooltip on the transferred annotations, there is a note saying that the annotation derives from the original sequence.

The **Codon Frequency Table** is used to determine the frequencies of the codons. Select a frequency table from the list that fits the organism you are working with. A translation table of an organism is created on the basis of counting all the codons in the coding sequences. Every codon in a **Codon Frequency Table** has its own count, frequency (per thousand) and fraction which are calculated in accordance with the occurrences of the codon in the organism. You can customize the list of codon frequency tables for your installation, see section L.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The newly created nucleotide sequence is shown, and if the analysis was performed on several protein sequences, there will be a corresponding number of views of nucleotide sequences. The new sequence is not saved automatically. To save the sequence, drag it into the **Navigation Area** or press Ctrl + S (⌘ + S on Mac) to show the save dialog.

16.9.2 Bioinformatics explained: Reverse translation

In all living cells containing hereditary material such as DNA, a transcription to mRNA and subsequent a translation to proteins occur. This is of course simplified but is in general what is happening in order to have a steady production of proteins needed for the survival of the cell. In bioinformatics analysis of proteins it is sometimes useful to know the ancestral DNA sequence in order to find the genomic localization of the gene. Thus, the translation of proteins back to DNA/RNA is of particular interest, and is called reverse translation or back-translation.

The Genetic Code

In 1968 the Nobel Prize in Medicine was awarded to Robert W. Holley, Har Gobind Khorana and Marshall W. Nirenberg for their interpretation of the Genetic Code (http://nobelprize.org/medicine/laureates/1968/). The Genetic Code represents translations of all 64 different codons into 20 different amino acids. Therefore it is no problem to translate a DNA/RNA sequence into a specific protein. But due to the degeneracy of the genetic code, several codons may code for only one specific amino acid. This can be seen in the table below. After the discovery of the genetic code it has been concluded that different organism (and organelles) have genetic codes which are different from the "standard genetic code". Moreover, the amino acid alphabet is no longer limited to 20 amino acids. The 21'st amino acid, selenocysteine, is encoded by an 'UGA' codon which is normally a stop codon. The discrimination of a selenocysteine over a stop codon is carried out by the translation machinery. Selenocysteines are very rare amino acids.

The table below shows the Standard Genetic Code which is the default translation table.

TTT F Phe	TCT S Ser	TAT Y Tyr	TGT C Cys
TTC F Phe	TCC S Ser	TAC Y Tyr	TGC C Cys
TTA L Leu	TCA S Ser	TAA * Ter	TGA * Ter
TTG L Leu i	TCG S Ser	TAG * Ter	TGG W Trp
CTT L Leu	CCT P Pro	CAT H His	CGT R Arg
CTC L Leu	CCC P Pro	CAC H His	CGC R Arg
CTA L Leu	CCA P Pro	CAA Q Gln	CGA R Arg
CTG L Leu i	CCG P Pro	CAG Q Gln	CGG R Arg
ATT I Ile	ACT T Thr	AAT N Asn	AGT S Ser
ATC I Ile	ACC T Thr	AAC N Asn	AGC S Ser
ATA I Ile	ACA T Thr	AAA K Lys	AGA R Arg
ATG M Met i	ACG T Thr	AAG K Lys	AGG R Arg
GTT V Val	GCT A Ala	GAT D Asp	GGT G Gly
GTC V Val	GCC A Ala	GAC D Asp	GGC G Gly
GTA V Val	GCA A Ala	GAA E Glu	GGA G Gly
GTG V Val	GCG A Ala	GAG E Glu	GGG G Gly

Challenge of reverse translation

A particular protein follows from the translation of a DNA sequence whereas the reverse translation need not have a specific solution according to the Genetic Code. The Genetic Code is degenerate which means that a particular amino acid can be translated into more than one codon. Hence there are ambiguities of the reverse translation.

Solving the ambiguities of reverse translation

In order to solve these ambiguities of reverse translation you can define how to prioritize the codon selection, e.g:

- Choose a codon randomly.
- Select the most frequent codon in a given organism.
- Randomize a codon, but with respect to its frequency in the organism.

As an example we want to translate an alanine to the corresponding codon. Four different codons can be used for this reverse translation; GCU, GCC, GCA or GCG. By picking either one by random choice we will get an alanine.

The most frequent codon, coding for an alanine in *E. coli* is GCG, encoding 33.7% of all alanines. Then comes GCC (25.5%), GCA (20.3%) and finally GCU (15.3%). The data are retrieved from the Codon usage database, see below. Always picking the most frequent codon does not necessarily give the best answer.

By selecting codons from a distribution of calculated codon frequencies, the DNA sequence obtained after the reverse translation, holds the correct (or nearly correct) codon distribution. It should be kept in mind that the obtained DNA sequence is not necessarily identical to the original one encoding the protein in the first place, due to the degeneracy of the genetic code.

In order to obtain the best possible result of the reverse translation, one should use the codon frequency table from the correct organism or a closely related species. The codon usage of the mitochondrial chromosome are often different from the native chromosome(s), thus mitochondrial codon frequency tables should only be used when working specifically with mitochondria.

Other useful resources

The Genetic Code at NCBI:
http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi?mode=c

Codon usage database:
http://www.kazusa.or.jp/codon/

Wikipedia on the genetic code
http://en.wikipedia.org/wiki/Genetic_code

Creative Commons License

16.10 Proteolytic cleavage detection

CLC Genomics Workbench offers to analyze protein sequences with respect to cleavage by a selection of proteolytic enzymes. This section explains how to adjust the detection parameters and offers basic information on proteolytic cleavage in general.

16.10.1 Proteolytic cleavage parameters

Given a protein sequence, *CLC Genomics Workbench* detects proteolytic cleavage sites in accordance with detection parameters and shows the detected sites as annotations on the sequence and in textual format in a table below the sequence view.

Detection of proteolytic cleavage sites is initiated by:

> **right-click a protein sequence in Navigation Area** | **Toolbox** | **Protein Analyses ( )** | **Proteolytic Cleavage, ( )**

This opens the dialog shown in figure 16.23:

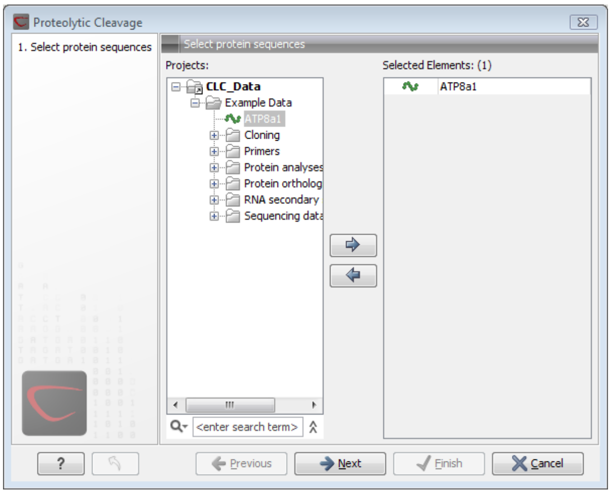
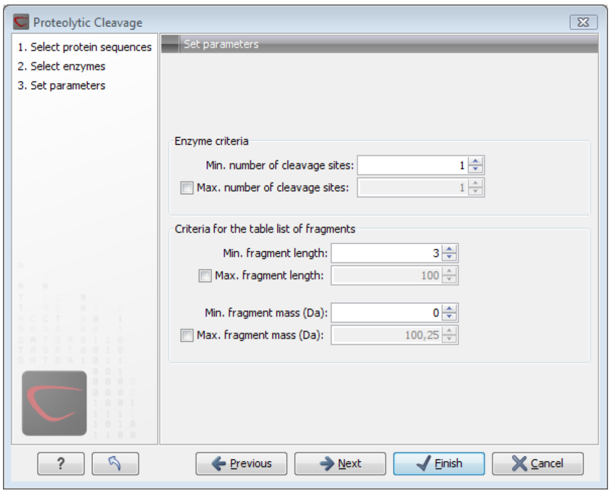


Figure 16.23: *Choosing sequence CAA32220 for proteolytic cleavage.*

CLC Genomics Workbench allows you to detect proteolytic cleavages for several sequences at a time. Correct the list of sequences by selecting a sequence and clicking the arrows pointing left and right. Then click **Next** to go to **Step 2**.

In **Step 2** you can select proteolytic cleavage enzymes. The list of available enzymes will be expanded continuously. Presently, the list contains the enzymes shown in figure 16.24. The full list of enzymes and their cleavage patterns can be seen in Appendix, section E.



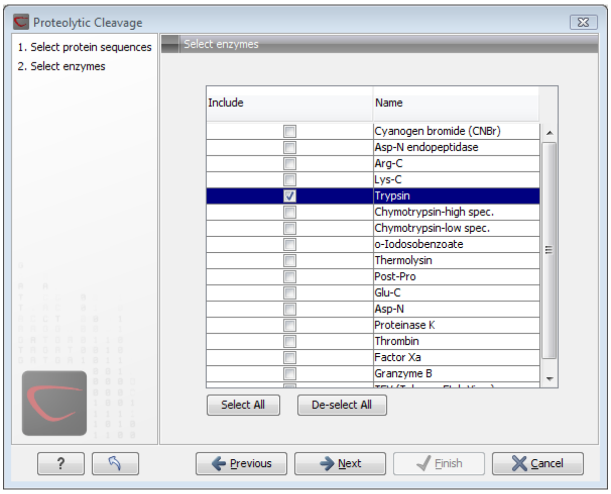
Min. number of cleavage sites: 1
□ Max. number of cleavage sites: 1
Min. fragment length: 3
□ Max. fragment length: 100
Min. fragment mass (Da): 0
□ Max. fragment mass (Da): 100,25

Figure 16.24: *Setting parameters for proteolytic cleavage detection.*

Select the enzymes you want to use for detection. When the relevant enzymes are chosen, click **Next**.

In **Step 3** you can set parameters for the detection. This limits the number of detected cleavages. Figure 16.25 shows an example of how parameters can be set.

- **Min. and max. number of cleavage sites.** Certain proteolytic enzymes cleave at many positions in the amino acid sequence. For instance proteinase K cleaves at nine different amino acids, regardless of the surrounding residues. Thus, it can be very useful to limit the number of actual cleavage sites before running the analysis.



Include	Name
□	Cyanogen bromide (CNBr)
□	Asp-N endopeptidase
□	Arg-C
□	Lys-C
☑	Trypsin
□	Chymotrypsin-high spec.
□	Chymotrypsin-low spec.
□	o-Iodosobenzoate
□	Thermolysin
□	Post-Pro
□	Glu-C
□	Asp-N
□	Proteinase K
□	Thrombin
□	Factor Xa
□	Granzyme B

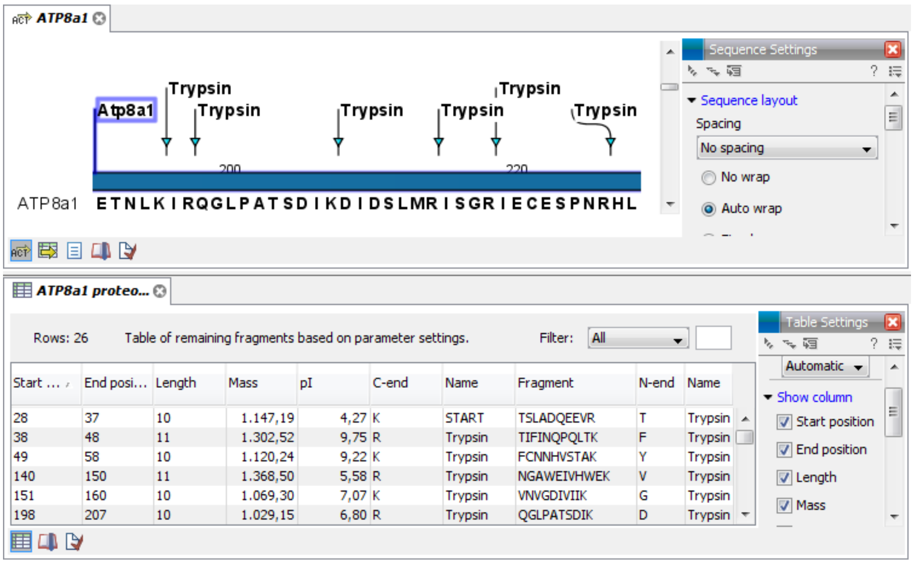
Figure 16.25: *Setting parameters for proteolytic cleavage detection.*

- **Min. and max. fragment length** Likewise, it is possible to limit the output to only display sequence fragments between a chosen length. Both a lower and upper limit can be chosen.
- **Min. and max. fragment mass** The molecular weight is not necessarily directly correlated to the fragment length as amino acids have different molecular masses. For that reason it is also possible to limit the search for proteolytic cleavage sites to mass-range.

Example!: If you have one protein sequence but you only want to show which enzymes cut between two and four times. Then you should select "The enzymes has more cleavage sites than 2" and select "The enzyme has less cleavage sites than 4". In the next step you should simply select all enzymes. This will result in a view where only enzymes which cut 2,3 or 4 times are presented.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

The result of the detection is displayed in figure 16.26.



Rows: 26 Table of remaining fragments based on parameter settings. Filter: All

Start ...	End posi...	Length	Mass	pI	C-end	Name	Fragment	N-end	Name
28	37	10	1.147,19	4,27	K	START	TSLADQEEVR	T	Trypsin
38	48	11	1.302,52	9,75	R	Trypsin	TIFINQPQLTK	F	Trypsin
49	58	10	1.120,24	9,22	K	Trypsin	FCNNHVSTAK	Y	Trypsin
140	150	11	1.368,50	5,58	R	Trypsin	NGAWEIVHWEK	V	Trypsin
151	160	10	1.069,30	7,07	K	Trypsin	VNVGDIVIIK	G	Trypsin
198	207	10	1.029,15	6,80	R	Trypsin	QGLPATSDIK	D	Trypsin

Figure 16.26: *The result of the proteolytic cleavage detection.*

Depending on the settings in the program, the output of the proteolytic cleavage site detection will display two views on the screen. The top view shows the actual protein sequence with the predicted cleavage sites indicated by small arrows. If no labels are found on the arrows they can be enabled by setting the labels in the "annotation layout" in the preference panel. The bottom view shows a text output of the detection, listing the individual fragments and information on these.

16.10.2 Bioinformatics explained: Proteolytic cleavage

Proteolytic cleavage is basically the process of breaking the peptide bonds between amino acids in proteins. This process is carried out by enzymes called peptidases, proteases or proteolytic cleavage enzymes.

Proteins often undergo proteolytic processing by specific proteolytic enzymes (proteases/peptidases) before final maturation of the protein. Proteins can also be cleaved as a result of intracellular processing of, for example, misfolded proteins. Another example of proteolytic processing of proteins is secretory proteins or proteins targeted to organelles, which have their signal peptide removed by specific signal peptidases before release to the extracellular environment or specific organelle.

Below a few processes are listed where proteolytic enzymes act on a protein substrate.

- N-terminal methionine residues are often removed after translation.
- Signal peptides or targeting sequences are removed during translocation through a membrane.
- Viral proteins that were translated from a monocistronic mRNA are cleaved.
- Proteins or peptides can be cleaved and used as nutrients.
- Precursor proteins are often processed to yield the mature protein.

Proteolytic cleavage of proteins has shown its importance in laboratory experiments where it is often useful to work with specific peptide fragments instead of entire proteins.

Proteases also have commercial applications. As an example proteases can be used as detergents for cleavage of proteinaceous stains in clothing.

The general nomenclature of cleavage site positions of the substrate were formulated by Schechter and Berger, 1967-68 [Schechter and Berger, 1967], [Schechter and Berger, 1968]. They designate the cleavage site between P1-P1', incrementing the numbering in the N-terminal direction of the cleaved peptide bond (P2, P3, P4, etc..). On the carboxyl side of the cleavage site the numbering is incremented in the same way (P1', P2', P3' etc.). This is visualized in figure 16.27.

Proteases often have a specific recognition site where the peptide bond is cleaved. As an example trypsin only cleaves at lysine or arginine residues, but it does not matter (with a few exceptions) which amino acid is located at position P1'(carboxyterminal of the cleavage site). Another example is trombin which cleaves if an arginine is found in position P1, but not if a D or E is found in position P1' at the same time. (See figure 16.28).

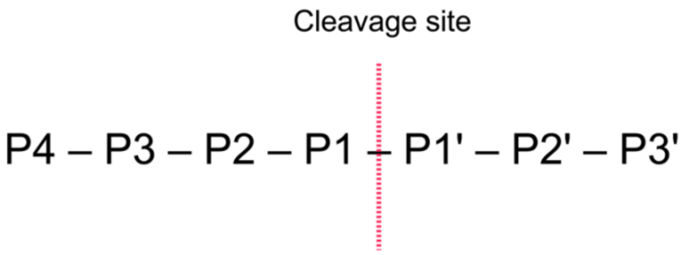


Figure 16.27: *Nomenclature of the peptide substrate. The substrate is cleaved between position P1-P1'.*

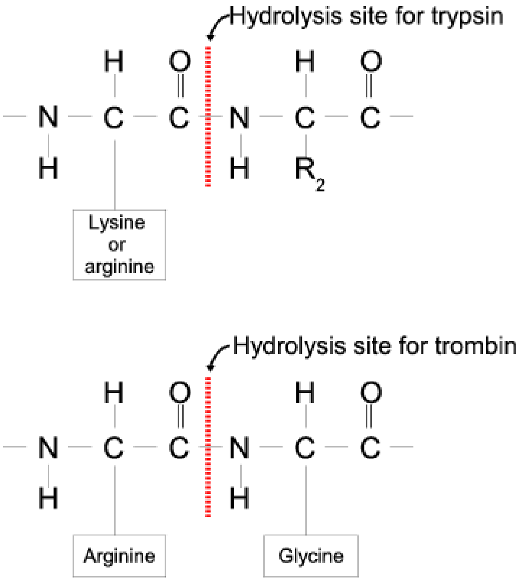


Figure 16.28: *Hydrolysis of the peptide bond between two amino acids. Trypsin cleaves unspecifically at lysine or arginine residues whereas trombin cleaves at arginines if asparate or glutamate is absent.*

Bioinformatics approaches are used to identify potential peptidase cleavage sites. Fragments can be found by scanning the amino acid sequence for patterns which match the corresponding cleavage site for the protease. When identifying cleaved fragments it is relatively important to know the calculated molecular weight and the isoelectric point.

Other useful resources

The Peptidase Database: http://merops.sanger.ac.uk/

Creative Commons License

Chapter 17

Primers

Contents

CLC Genomics Workbench offers graphically and algorithmically advanced design of primers and probes for various purposes. This chapter begins with a brief introduction to the general concepts of the primer designing process. Then follows instructions on how to adjust parameters for primers, how to inspect and interpret primer properties graphically and how to interpret, save and analyze the output of the primer design analysis. After a description of the different reaction types for which primers can be designed, the chapter closes with sections on how to match primers with other sequences and how to create a primer order.

17.1 Primer design - an introduction

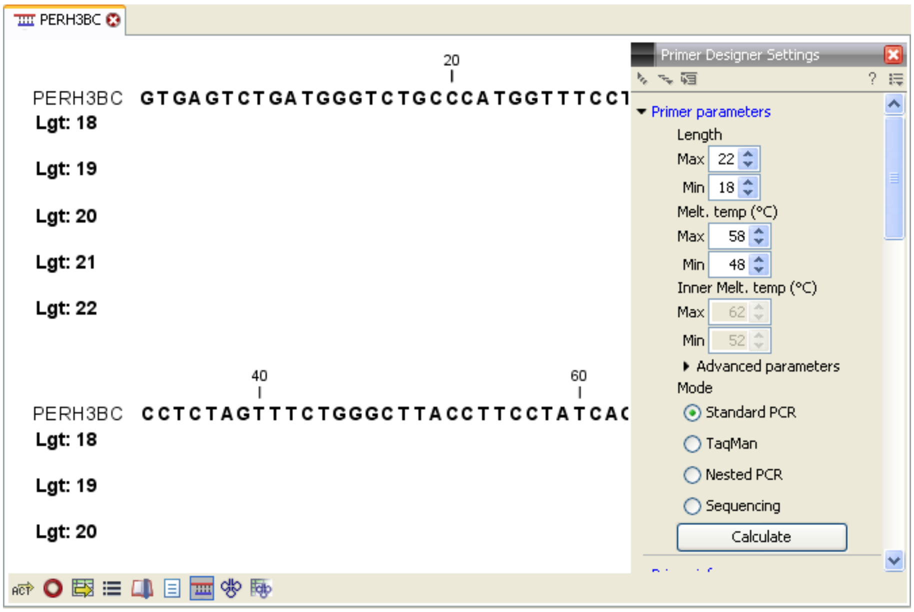
Primer design can be accessed in two ways:

select sequence | Toolbox in the Menu Bar | Primers and Probes ( ) | Design Primers ( ) | OK

or **right-click sequence | Show | Primer ( )**

In the primer view (see figure 17.1), the basic options for viewing the template sequence are the same as for the standard sequence view. See section 10.1 for an explanation of these options.

Note! This means that annotations such as e.g. known SNP's or exons can be displayed on the template sequence to guide the choice of primer regions. Also, traces in sequencing reads can be shown along with the structure to guide e.g. the re-sequencing of poorly resolved regions.



PERH3BC

```
                               20
PERH3BC  GTGAGTCTGATGGGTCTGCCCATGGTTTCCT
Lgt: 18

Lgt: 19

Lgt: 20

Lgt: 21

Lgt: 22

                  40                   60
PERH3BC  CCTCTAGTTTCTGGGCTTACCTTCCTATCAC
Lgt: 18

Lgt: 19

Lgt: 20
```

Primer Designer Settings

Primer parameters

Length
Max 22
Min 18

Melt. temp (°C)
Max 58
Min 48

Inner Melt. temp (°C)
Max 62
Min 52

Advanced parameters

Mode
● Standard PCR
○ TaqMan
○ Nested PCR
○ Sequencing

Figure 17.1: *The initial view of the sequence used for primer design.*

17.1.1 General concept

The concept of the primer view is that the user first chooses the desired reaction type for the session in the Primer Parameters preference group, e.g. *Standard PCR*. Reflecting the choice of reaction type, it is now possibly to select one or more regions on the sequence and to use the right-click mouse menu to designate these as primer or probe regions (see figure 17.2).

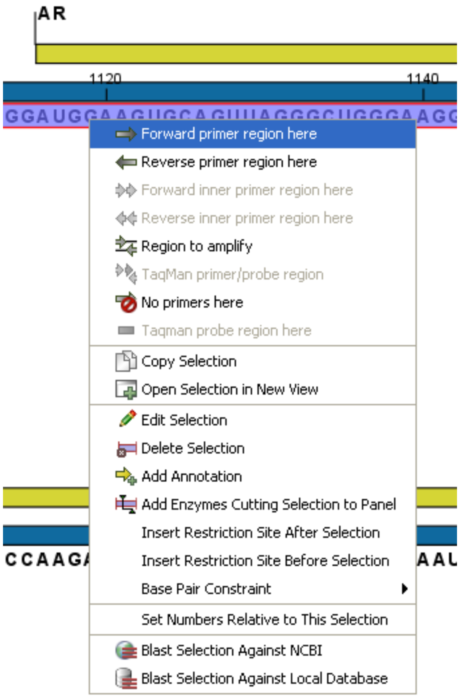

AR
1120
1140
GGAUGGAAGUGCAGUUAGGGCUGGGAAGG
CCAAGA
AAU
Forward primer region here
Reverse primer region here
Forward inner primer region here
Reverse inner primer region here
Region to amplify
TaqMan primer/probe region
No primers here
Taqman probe region here
Copy Selection
Open Selection in New View
Edit Selection
Delete Selection
Add Annotation
Add Enzymes Cutting Selection to Panel
Insert Restriction Site After Selection
Insert Restriction Site Before Selection
Base Pair Constraint
Set Numbers Relative to This Selection
Blast Selection Against NCBI
Blast Selection Against Local Database

Figure 17.2: *Right-click menu allowing you to specify regions for the primer design*

When a region is chosen, graphical information about the properties of all possible primers in this region will appear in lines beneath it. By default, information is showed using a compact mode but the user can change to a more detailed mode in the Primer information preference group.

The number of information lines reflects the chosen length interval for primers and probes. In the compact information mode one line is shown for every possible primer-length and each of these lines contain information regarding all possible primers of the given length. At each potential primer starting position, a circular information point is shown which indicates whether the primer fulfills the requirements set in the primer parameters preference group. A green circle indicates a primer which fulfils all criteria and a red circle indicates a primer which fails to meet one or more of the set criteria. For more detailed information, place the mouse cursor over the circle representing the primer of interest. A tool-tip will then appear on screen, displaying detailed information about the primer in relation to the set criteria. To locate the primer on the sequence, simply left-click the circle using the mouse.

The various primer parameters can now be varied to explore their effect and the view area will dynamically update to reflect this allowing for a high degree of interactivity in the primer design process.

After having explored the potential primers the user may have found a satisfactory primer and choose to export this directly from the view area using a mouse right-click on the primers information point. This does not allow for any design information to enter concerning the properties of primer/probe pairs or sets e.g. primer pair annealing and T_m difference between primers. If the latter is desired the user can use the **Calculate** button at the bottom of the Primer parameter preference group. This will activate a dialog, the contents of which depends on the chosen mode. Here, the user can set primer-pair specific setting such as allowed or desired T_m

difference and view the single-primer parameters which were chosen in the Primer parameters preference group.

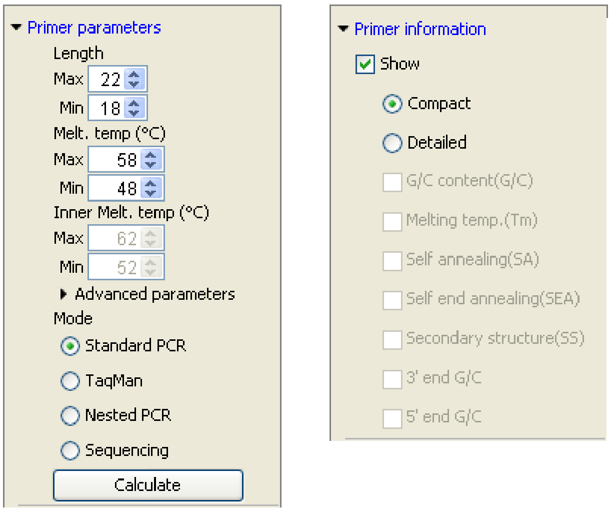
Upon pressing finish, an algorithm will generate all possible primer sets and rank these based on their characteristics and the chosen parameters. A list will appear displaying the 100 most high scoring sets and information pertaining to these. The search result can be saved to the navigator. From the result table, suggested primers or primer/probe sets can be explored since clicking an entry in the table will highlight the associated primers and probes on the sequence. It is also possible to save individual primers or sets from the table through the mouse right-click menu. For a given primer pair, the amplified PCR fragment can also be opened or saved using the mouse right-click menu.

17.1.2 Scoring primers

CLC Genomics Workbench employs a proprietary algorithm to rank primer and probe solutions. The algorithm considers both the parameters pertaining to single oligos, such as e.g. the secondary structure score and parameters pertaining to oligo-pairs such as e.g. the oligo pair-annealing score. The ideal score for a solution is 100 and solutions are thus ranked in descending order. Each parameter is assigned an ideal value and a tolerance. Consider for example oligo self-annealing, here the ideal value of the annealing score is 0 and the tolerance corresponds to the maximum value specified in the side panel. The contribution to the final score is determined by how much the parameter deviates from the ideal value and is scaled by the specified tolerance. Hence, a large deviation from the ideal and a small tolerance will give a large deduction in the final score and a small deviation from the ideal and a high tolerance will give a small deduction in the final score.

17.2 Setting parameters for primers and probes

The primer-specific view options and settings are found in the **Primer parameters** preference group in the **Side Panel** to the right of the view (see figure 17.3).



Primer parameters
Length
Max 22
Min 18
Melt. temp (°C)
Max 58
Min 48
Inner Melt. temp (°C)
Max 62
Min 52
Advanced parameters
Mode
● Standard PCR
○ TaqMan
○ Nested PCR
○ Sequencing
Calculate

Primer information
☑ Show
● Compact
○ Detailed
□ G/C content(G/C)
□ Melting temp.(Tm)
□ Self annealing(SA)
□ Self end annealing(SEA)
□ Secondary structure(SS)
□ 3' end G/C
□ 5' end G/C

Figure 17.3: *The two groups of primer parameters (in the program, the Primer information group is listed below the other group).*

17.2.1 Primer Parameters

In this preference group a number of criteria can be set, which the selected primers must meet. All the criteria concern *single primers*, as primer pairs are not generated until the **Calculate** button is pressed. Parameters regarding primer and probe sets are described in detail for each reaction mode (see below).

- **Length.** Determines the length interval within which primers can be designed by setting a maximum and a minimum length. The upper and lower lengths allowed by the program are 50 and 10 nucleotides respectively.
- **Melting temperature.** Determines the temperature interval within which primers must lie. When the *Nested PCR* or *TaqMan* reaction type is chosen, the first pair of melting temperature interval settings relate to the outer primer pair i.e. not the probe. Melting temperatures are calculated by a nearest-neighbor model which considers stacking interactions between neighboring bases in the primer-template complex. The model uses state-of-the-art thermodynamic parameters [SantaLucia, 1998] and considers the important contribution from the dangling ends that are present when a short primer anneals to a template sequence [Bommarito et al., 2000]. A number of parameters can be adjusted concerning the reaction mixture and which influence melting temperatures (see below). Melting temperatures are corrected for the presence of monovalent cations using the model of [SantaLucia, 1998] and temperatures are further corrected for the presence of magnesium, deoxynucleotide triphosphates (dNTP) and dimethyl sulfoxide (DMSO) using the model of [von Ahsen et al., 2001].
- **Inner melting temperature.** This option is only activated when the *Nested PCR* or *TaqMan* mode is selected. In *Nested PCR* mode, it determines the allowed melting temperature interval for the inner/nested pair of primers, and in *TaqMan* mode it determines the allowed temperature interval for the TaqMan probe.
- **Advanced parameters.** A number of less commonly used options
  - **Buffer properties.** A number of parameters concerning the reaction mixture which influence melting temperatures.
    - **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM)
    - **Salt concentration.** Specifies the concentration of monovalent cations ($[NA^{+}]$, $[K^{+}]$ and equivalents) in units of millimoles (mM)
    - **Magnesium concentration.** Specifies the concentration of magnesium cations ($[Mg^{++}]$) in units of millimoles (mM)
    - **dNTP concentration.** Specifies the concentration of deoxynucleotide triphosphates in units of millimoles (mM)
    - **DMSO concentration.** Specifies the concentration of dimethyl sulfoxide in units of volume percent $(vol.\%)$
  - **GC content.** Determines the interval of CG content (% C and G nucleotides in the primer) within which primers must lie by setting a maximum and a minimum GC content.
  - **Self annealing.** Determines the maximum self annealing value of all primers and probes. This determines the amount of base-pairing allowed between two copies of

the same molecule. The self annealing score is measured in number of hydrogen bonds between two copies of primer molecules, with A-T base pairs contributing 2 hydrogen bonds and G-C base pairs contributing 3 hydrogen bonds.

- **Self end annealing.** Determines the maximum self end annealing value of all primers and probes. This determines the number of consecutive base pairs allowed between the 3' end of one primer and another copy of that primer. This score is calculated in number of hydrogen bonds (the example below has a score of 4 - derived from 2 A-T base pairs each with 2 hydrogen bonds).

```
AATTCCCTACAATCCCCAAA
                  ||
  AAACCCCTAACATCCCTTAA
```

.

- **Secondary structure.** Determines the maximum score of the optimal secondary DNA structure found for a primer or probe. Secondary structures are scored by the number of hydrogen bonds in the structure, and 2 extra hydrogen bonds are added for each stacking base-pair in the structure.

- **3' end G/C restrictions.** When this checkbox is selected it is possible to specify restrictions concerning the number of G and C molecules in the 3' end of primers and probes. A low G/C content of the primer/probe 3' end increases the specificity of the reaction. A high G/C content facilitates a tight binding of the oligo to the template but also increases the possibility of mispriming. Unfolding the preference groups yields the following options:
  - **End length.** The number of consecutive terminal nucleotides for which to consider the C/G content
  - **Max no. of G/C.** The maximum number of G and C nucleotides allowed within the specified length interval
  - **Min no. of G/C.** The minimum number of G and C nucleotides required within the specified length interval

- **5' end G/C restrictions.** When this checkbox is selected it is possible to specify restrictions concerning the number of G and C molecules in the 5' end of primers and probes. A high G/C content facilitates a tight binding of the oligo to the template but also increases the possibility of mis-priming. Unfolding the preference groups yields the same options as described above for the 3' end.

- **Mode.** Specifies the reaction type for which primers are designed:
  - **Standard PCR.** Used when the objective is to design primers, or primer pairs, for PCR amplification of a single DNA fragment.
  - **Nested PCR.** Used when the objective is to design two primer pairs for nested PCR amplification of a single DNA fragment.
  - **Sequencing.** Used when the objective is to design primers for DNA sequencing.
  - **TaqMan.** Used when the objective is to design a primer pair and a probe for TaqMan quantitative PCR.

  Each mode is described further below.

- **Calculate.** Pushing this button will activate the algorithm for designing primers

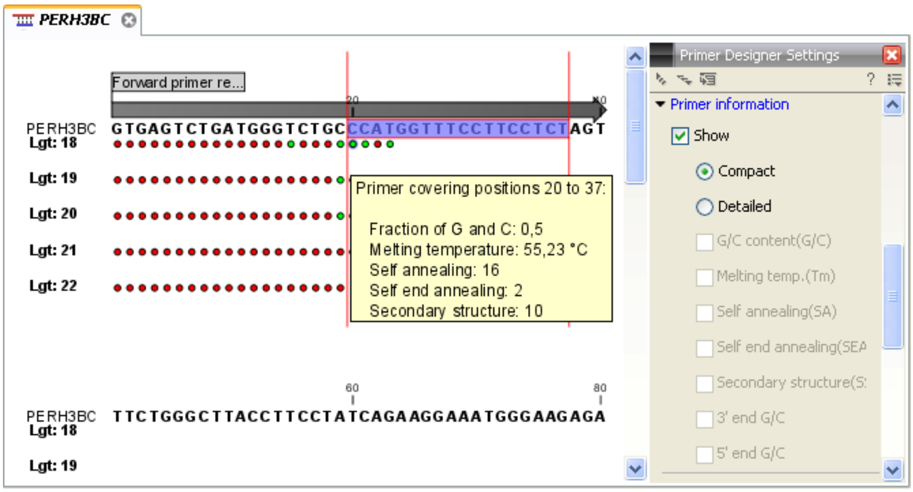
17.3 Graphical display of primer information

The primer information settings are found in the **Primer information** preference group in the **Side Panel** to the right of the view (see figure 17.3).

There are two different ways to display the information relating to a single primer, the detailed and the compact view. Both are shown below the primer regions selected on the sequence.

17.3.1 Compact information mode

This mode offers a condensed overview of all the primers that are available in the selected region. When a region is chosen primer information will appear in lines beneath it (see figure 17.4).



Primer covering positions 20 to 37:

Fraction of G and C: 0,5
Melting temperature: 55,23 °C
Self annealing: 16
Self end annealing: 2
Secondary structure: 10

Primer information

☑ Show

● Compact

○ Detailed

□ G/C content(G/C)

□ Melting temp.(Tm)

□ Self annealing(SA)

□ Self end annealing(SEA

□ Secondary structure(S:

□ 3' end G/C

□ 5' end G/C

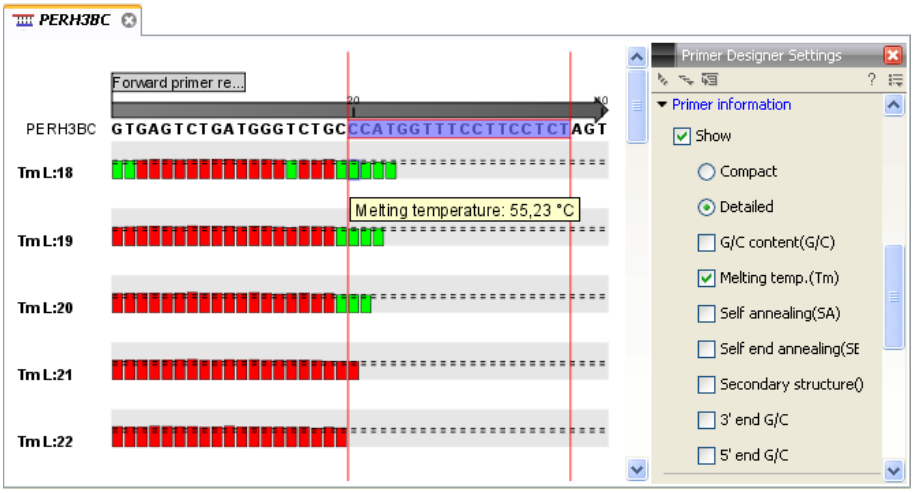
Figure 17.4: *Compact information mode*

The number of information lines reflects the chosen length interval for primers and probes. One line is shown for every possible primer-length, if the length interval is widened more lines will appear. At each potential primer starting position a circle is shown which indicates whether the primer fulfills the requirements set in the primer parameters preference group. A green primer indicates a primer which fulfils all criteria and a red primer indicates a primer which fails to meet one or more of the set criteria. For more detailed information, place the mouse cursor over the circle representing the primer of interest. A tool-tip will then appear on screen displaying detailed information about the primer in relation to the set criteria. To locate the primer on the sequence, simply left-click the circle using the mouse.

The various primer parameters can now be varied to explore their effect and the view area will dynamically update to reflect this. If e.g. the allowed melting temperature interval is widened more green circles will appear indicating that more primers now fulfill the set requirements and if e.g. a requirement for 3' G/C content is selected, rec circles will appear at the starting points of the primers which fail to meet this requirement.

17.3.2 Detailed information mode

In this mode a very detailed account is given of the properties of all the available primers. When a region is chosen primer information will appear in groups of lines beneath it (see figure 17.5).


PERH3BC
Forward primer re...
20
40
PERH3BC GTGAGTCTGATGGGTCTGCCCATGGTTTCCTTCCTCTAGT
Tm L:18
Tm L:19
Tm L:20
Tm L:21
Tm L:22
Melting temperature: 55,23 °C

Primer Designer Settings

Primer information

☑ Show

○ Compact

● Detailed

□ G/C content(G/C)

☑ Melting temp.(Tm)

□ Self annealing(SA)

□ Self end annealing(SE

□ Secondary structure()

□ 3' end G/C

□ 5' end G/C

Figure 17.5: *Detailed information mode*

The number of information-line-groups reflects the chosen length interval for primers and probes. One group is shown for every possible primer length. Within each group, a line is shown for every primer property that is selected from the checkboxes in the primer information preference group. Primer properties are shown at each potential primer starting position and are of two types:

Properties with numerical values are represented by bar plots. A green bar represents the starting point of a primer that meets the set requirement and a red bar represents the starting point of a primer that fails to meet the set requirement:

- G/C content
- Melting temperature
- Self annealing score
- Self end annealing score
- Secondary structure score

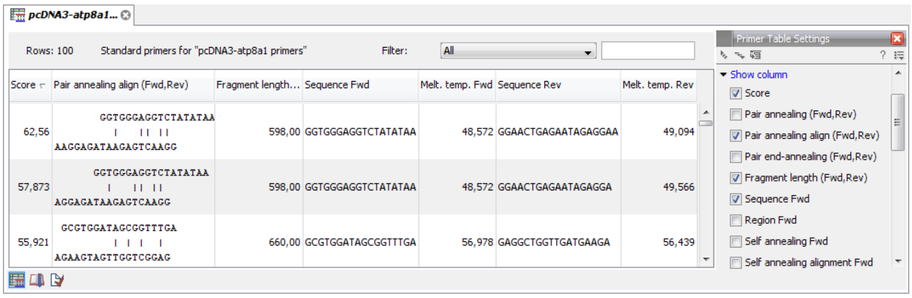
Properties with Yes - No values. If a primer meets the set requirement a green circle will be shown at its starting position and if it fails to meet the requirement a red dot is shown at its starting position:

- C/G at 3' end
- C/G at 5' end

Common to both sorts of properties is that mouse clicking an information point (filled circle or bar) will cause the region covered by the associated primer to be selected on the sequence.

17.4 Output from primer design

The output generated by the primer design algorithm is a table of proposed primers or primer pairs with the accompanying information (see figure 17.6).



pcDNA3-atp8a1...

Rows: 100 Standard primers for "pcDNA3-atp8a1 primers" Filter: All

<table>
<tr><th>Score</th><th>Pair annealing align (Fwd,Rev)</th><th>Fragment length...</th><th>Sequence Fwd</th><th>Melt. temp. Fwd</th><th>Sequence Rev</th><th>Melt. temp. Rev</th></tr>
<tr><td>62,56</td><td>GGTGGGAGGTCTATATAA
| || ||
AAGGAGATAAGAGTCAAGG</td><td>598,00</td><td>GGTGGGAGGTCTATATAA</td><td>48,572</td><td>GGAACTGAGAATAGAGGAA</td><td>49,094</td></tr>
<tr><td>57,873</td><td>GGTGGGAGGTCTATATAA
| || ||
AGGAGATAAGAGTCAAGG</td><td>598,00</td><td>GGTGGGAGGTCTATATAA</td><td>48,572</td><td>GGAACTGAGAATAGAGGA</td><td>49,566</td></tr>
<tr><td>55,921</td><td>GCGTGGATAGCGGTTTGA
| | | |
AGAAGTAGTTGGTCGGAG</td><td>660,00</td><td>GCGTGGATAGCGGTTTGA</td><td>56,978</td><td>GAGGCTGGTTGATGAAGA</td><td>56,439</td></tr>
</table>

Primer Table Settings

Show column

- ☑ Score
- ☐ Pair annealing (Fwd,Rev)
- ☑ Pair annealing align (Fwd,Rev)
- ☐ Pair end-annealing (Fwd,Rev)
- ☑ Fragment length (Fwd,Rev)
- ☑ Sequence Fwd
- ☐ Region Fwd
- ☐ Self annealing Fwd
- ☐ Self annealing alignment Fwd

Figure 17.6: *Proposed primers*

In the preference panel of the table, it is possible to customize which columns are shown in the table. See the sections below on the different reaction types for a description of the available information.

The columns in the output table can be sorted by the present information. For example the user can choose to sort the available primers by their score (default) or by their self annealing score, simply by right-clicking the column header.

The output table interacts with the accompanying primer editor such that when a proposed combination of primers and probes is selected in the table the primers and probes in this solution are highlighted on the sequence.

17.4.1 Saving primers

Primer solutions in a table row can be saved by selecting the row and using the right-click mouse menu. This opens a dialog that allows the user to save the primers to the desired location. Primers and probes are saved as DNA sequences in the program. This means that all available DNA analyzes can be performed on the saved primers, including BLAST. Furthermore, the primers can be edited using the standard sequence view to introduce e.g. mutations and restriction sites.

17.4.2 Saving PCR fragments

The PCR fragment generated from the primer pair in a given table row can also be saved by selecting the row and using the right-click mouse menu. This opens a dialog that allows the user to save the fragment to the desired location. The fragment is saved as a DNA sequence and the position of the primers is added as annotation on the sequence. The fragment can then be used for further analysis and included in e.g. an in-silico cloning experiment using the cloning editor.

17.4.3 Adding primer binding annotation

You can add an annotation to the template sequence specifying the binding site of the primer: Right-click the primer in the table and select **Mark primer annotation on sequence**.

17.5 Standard PCR

This mode is used to design primers for a PCR amplification of a single DNA fragment.

17.5.1 User input

In this mode the user must define either a *Forward primer region*, a *Reverse primer region*, or both. These are defined by making a selection on the sequence and right-clicking the selection. It is also possible to define a *Region to amplify* in which case a forward- and a reverse primer region are automatically placed so as to ensure that the designated region will be included in the PCR fragment. If areas are known where primers must not bind (e.g. repeat rich areas), one or more *No primers here* regions can be defined.

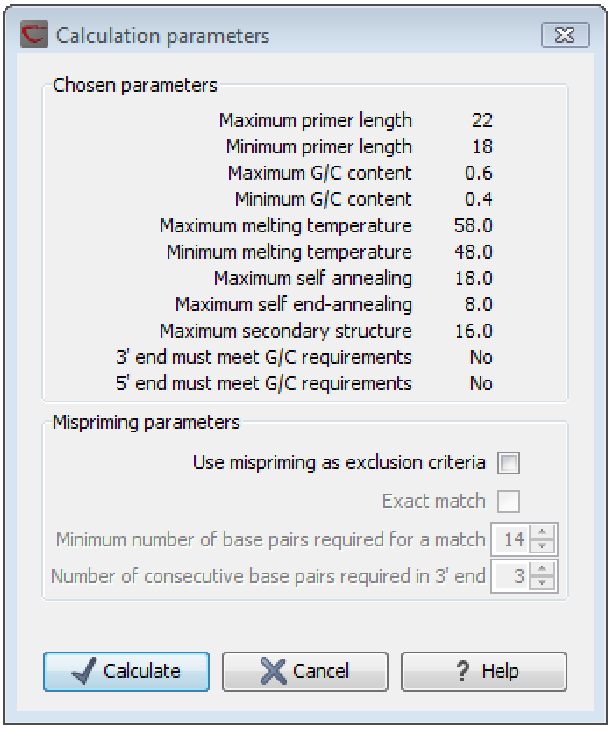
If two regions are defined, it is required that at least a part of the *Forward primer region* is located upstream of the *Reverse primer region*.

After exploring the available primers (see section 17.3) and setting the desired parameter values in the Primer Parameters preference group, the **Calculate** button will activate the primer design algorithm.

When a single primer region is defined

If only a single region is defined, only *single primers* will be suggested by the algorithm.

After pressing the **Calculate** button a dialog will appear (see figure 17.7).



Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Mispriming parameters

Use mispriming as exclusion criteria □

Exact match □

Minimum number of base pairs required for a match 14

Number of consecutive base pairs required in 3' end 3

Figure 17.7: *Calculation dialog for PCR primers when only a single primer region has been defined.*

The top part of this dialog shows the parameter settings chosen in the Primer parameters preference group which will be used by the design algorithm.

The lower part contains a menu where the user can choose to include mispriming as a criteria in the design process. If this option is selected the algorithm will search for competing binding sites of the primer within the sequence.

The adjustable parameters for the search are:

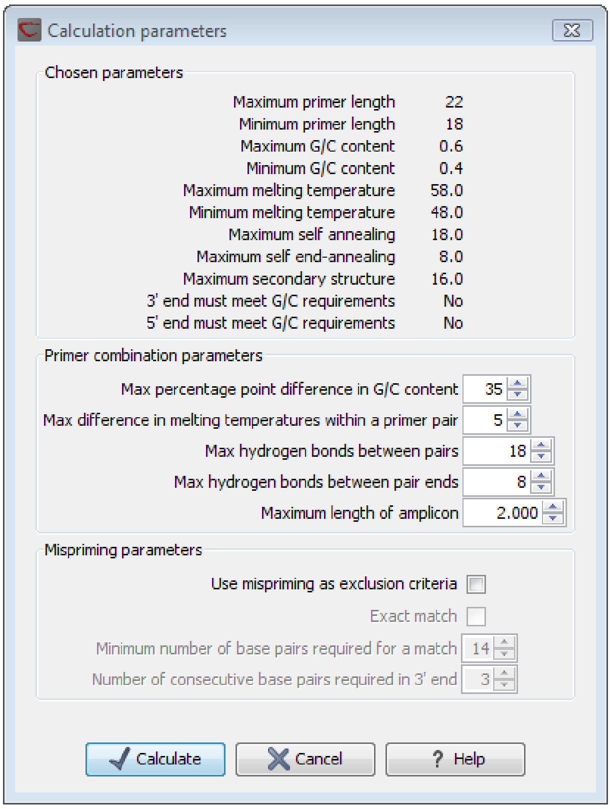
- **Exact match**. Choose only to consider exact matches of the primer, i.e. all positions must base pair with the template for mispriming to occur.
- **Minimum number of base pairs required for a match**. How many nucleotides of the primer that must base pair to the sequence in order to cause mispriming.
- **Number of consecutive base pairs required in 3' end**. How many consecutive 3' end base pairs in the primer that MUST be present for mispriming to occur. This option is included since 3' terminal base pairs are known to be essential for priming to occur.

Note! Including a search for potential mispriming sites will prolong the search time substantially if long sequences are used as template and if the minimum number of base pairs required for a match is low. If the region to be amplified is part of a very long molecule and mispriming is a concern, consider extracting part of the sequence prior to designing primers.

When both forward and reverse regions are defined

If both a forward and a reverse region are defined, *primer pairs* will be suggested by the algorithm.

After pressing the **Calculate** button a dialog will appear (see figure 17.8).



Calculation parameters

Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Primer combination parameters

Max percentage point difference in G/C content	35
Max difference in melting temperatures within a primer pair	5
Max hydrogen bonds between pairs	18
Max hydrogen bonds between pair ends	8
Maximum length of amplicon	2.000

Mispriming parameters

Use mispriming as exclusion criteria □

Exact match □

Minimum number of base pairs required for a match 14

Number of consecutive base pairs required in 3' end 3

Figure 17.8: *Calculation dialog for PCR primers when two primer regions have been defined.*

Again, the top part of this dialog shows the parameter settings chosen in the Primer parameters preference group which will be used by the design algorithm. The lower part again contains a menu where the user can choose to include mispriming of both primers as a criteria in the design process (see above). The central part of the dialog contains parameters pertaining to primer pairs. Here three parameters can be set:

- Maximum percentage point difference in G/C content - if this is set at e.g. 5 points a pair of primers with 45% and 49% G/C nucleotides, respectively, will be allowed, whereas a pair of primers with 45% and 51% G/C nucleotides, respectively will not be included.
- Maximal difference in melting temperature of primers in a pair - the number of degrees Celsius that primers in a pair are all allowed to differ.
- Max hydrogen bonds between pairs - the maximum number of hydrogen bonds allowed between the forward and the reverse primer in a primer pair.
- Max hydrogen bonds between pair ends - the maximum number of hydrogen bonds allowed in the consecutive ends of the forward and the reverse primer in a primer pair.
- Maximum length of amplicon - determines the maximum length of the PCR fragment.

17.5.2 Standard PCR output table

If only a single region is selected the following columns of information are available:

- Sequence - the primer's sequence.
- Score - measures how much the properties of the primer (or primer pair) deviates from the optimal solution in terms of the chosen parameters and tolerances. The higher the score, the better the solution.
- Region - the interval of the template sequence covered by the primer
- Self annealing - the maximum self annealing score of the primer in units of hydrogen bonds
- Self annealing alignment - a visualization of the highest maximum scoring self annealing alignment
- Self end annealing - the maximum score of consecutive end base-pairings allowed between the ends of two copies of the same molecule in units of hydrogen bonds
- GC content - the fraction of G and C nucleotides in the primer
- Melting temperature of the primer-template complex
- Secondary structure score - the score of the optimal secondary DNA structure found for the primer. Secondary structures are scored by adding the number of hydrogen bonds in the structure, and 2 extra hydrogen bonds are added for each stacking base-pair in the structure
- Secondary structure - a visualization of the optimal DNA structure found for the primer

If both a forward and a reverse region are selected a table of primer pairs is shown, where the above columns (excluding the score) are represented twice, once for the forward primer (designated by the letter F) and once for the reverse primer (designated by the letter R).

Before these, and following the score of the primer pair, are the following columns pertaining to primer pair-information available:

- Pair annealing - the number of hydrogen bonds found in the optimal alignment of the forward and the reverse primer in a primer pair
- Pair annealing alignment - a visualization of the optimal alignment of the forward and the reverse primer in a primer pair.
- Pair end annealing - the maximum score of consecutive end base-pairings found between the ends of the two primers in the primer pair, in units of hydrogen bonds
- Fragment length - the length (number of nucleotides) of the PCR fragment generated by the primer pair

17.6 Nested PCR

Nested PCR is a modification of Standard PCR, aimed at reducing product contamination due to the amplification of unintended primer binding sites (mispriming). If the intended fragment can not be amplified without interference from competing binding sites, the idea is to seek out a larger outer fragment which can be unambiguously amplified and which contains the smaller intended fragment. Having amplified the outer fragment to large numbers, the PCR amplification of the inner fragment can proceed and will yield amplification of this with minimal contamination.

Primer design for nested PCR thus involves designing two primer pairs, one for the outer fragment and one for the inner fragment.

In *Nested PCR* mode the user must thus define four regions a *Forward primer region* (the outer forward primer), a *Reverse primer region* (the outer reverse primer), a *Forward inner primer region*, and a *Reverse inner primer region*. These are defined by making a selection on the sequence and right-clicking the selection. If areas are known where primers must not bind (e.g. repeat rich areas), one or more *No primers here* regions can be defined.

It is required that the *Forward primer region*, is located upstream of the *Forward inner primer region*, that the *Forward inner primer region*, is located upstream of the *Reverse inner primer region*, and that the *Reverse inner primer region*, is located upstream of the *Reverse primer region*.

In *Nested PCR* mode the *Inner melting temperature* menu in the Primer parameters panel is activated, allowing the user to set a separate melting temperature interval for the inner and outer primer pairs.

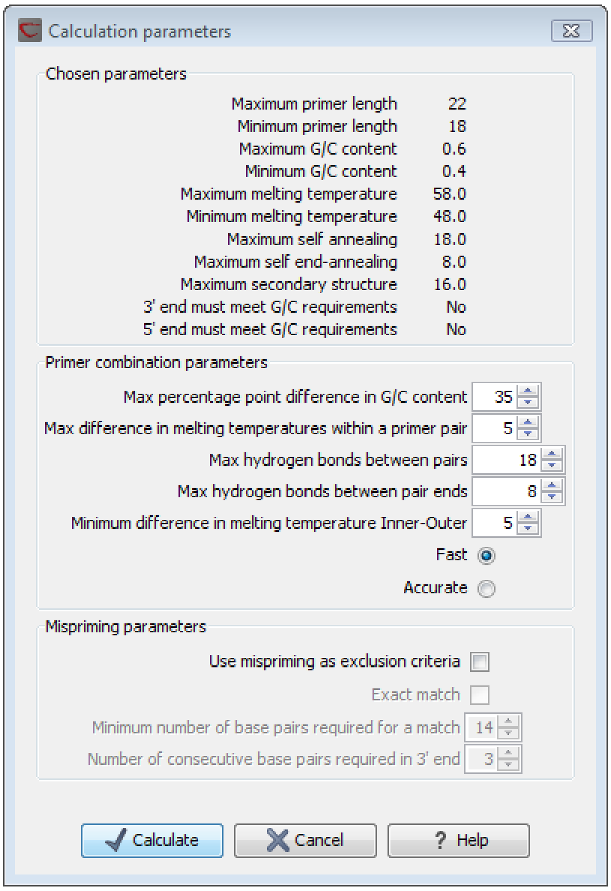
After exploring the available primers (see section 17.3) and setting the desired parameter values in the Primer parameters preference group, the **Calculate** button will activate the primer design algorithm.

After pressing the **Calculate** button a dialog will appear (see figure 17.9).

The top and bottom parts of this dialog are identical to the *Standard PCR* dialog for designing primer pairs described above.

The central part of the dialog contains parameters pertaining to primer pairs and the comparison between the outer and the inner pair. Here five options can be set:

- Maximum percentage point difference in G/C content (described above under Standard PCR) - this criteria is applied to both primer pairs independently.



Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Primer combination parameters

Max percentage point difference in G/C content	35
Max difference in melting temperatures within a primer pair	5
Max hydrogen bonds between pairs	18
Max hydrogen bonds between pair ends	8
Minimum difference in melting temperature Inner-Outer	5
Fast	●
Accurate	○

Mispriming parameters

Use mispriming as exclusion criteria	□
Exact match	□
Minimum number of base pairs required for a match	14
Number of consecutive base pairs required in 3' end	3

Figure 17.9: *Calculation dialog*

- Maximal difference in melting temperature of primers in a pair - the number of degrees Celsius that primers in a pair are all allowed to differ. This criteria is applied to both primer pairs independently.

- Maximum pair annealing score - the maximum number of hydrogen bonds allowed between the forward and the reverse primer in a primer pair. This criteria is applied to all possible combinations of primers.

- Minimum difference in the melting temperature of primers in the inner and outer primer pair - all comparisons between the melting temperature of primers from the two pairs must be at least this different, otherwise the primer set is excluded. This option is applied to ensure that the inner and outer PCR reactions can be initiated at different annealing temperatures. Please note that to ensure flexibility there is no directionality indicated when setting parameters for melting temperature differences between inner and outer primer pair, i.e. it is not specified whether the inner pair should have a lower or higher T_m. Instead this is determined by the allowed temperature intervals for inner and outer primers that are set in the primer parameters preference group in the side panel. If a higher T_m of inner primers is desired, choose a T_m interval for inner primers which has higher values than the interval for outer primers.

- Two radio buttons allowing the user to choose between a fast and an accurate algorithm for primer prediction.

17.6.1 Nested PCR output table

In nested PCR there are four primers in a solution, forward outer primer (FO), forward inner primer (FI), reverse inner primer (RI) and a reverse outer primer (RO).

The output table can show primer-pair combination parameters for all four combinations of primers and single primer parameters for all four primers in a solution (see section on Standard PCR for an explanation of the available primer-pair and single primer information).

The fragment length in this mode refers to the length of the PCR fragment generated by the inner primer pair, and this is also the PCR fragment which can be exported.

17.7 TaqMan

CLC Genomics Workbench allows the user to design primers and probes for TaqMan PCR applications.

TaqMan probes are oligonucleotides that contain a fluorescent reporter dye at the 5' end and a quenching dye at the 3' end. Fluorescent molecules become excited when they are irradiated and usually emit light. However, in a TaqMan probe the energy from the fluorescent dye is transferred to the quencher dye by fluorescence resonance energy transfer as long as the quencher and the dye are located in close proximity i.e. when the probe is intact. TaqMan probes are designed to anneal within a PCR product amplified by a standard PCR primer pair. If a TaqMan probe is bound to a product template, the replication of this will cause the Taq polymerase to encounter the probe. Upon doing so, the 5'exonuclease activity of the polymerase will cleave the probe. This cleavage separates the quencher and the dye, and as a result the reporter dye starts to emit fluorescence.

The TaqMan technology is used in Real-Time quantitative PCR. Since the accumulation of fluorescence mirrors the accumulation of PCR products it can can be monitored in real-time and used to quantify the amount of template initially present in the buffer.

The technology is also used to detect genetic variation such as SNP's. By designing a TaqMan probe which will specifically bind to one of two or more genetic variants it is possible to detect genetic variants by the presence or absence of fluorescence in the reaction.

Note! In *CLC Genomics Workbench* it is possible to annotate sequences with SNP information from dbSNP and use this information to guide TaqMan allele-specific probe design.

A specific requirement of TaqMan probes is that a G nucleotide can not be present at the 5' end since this will quench the fluorescence of the reporter dye. It is recommended that the melting temperature of the TaqMan probe is about 10 degrees celsius higher than that of the primer pair.

Primer design for TaqMan technology involves designing a primer pair and a TaqMan probe.

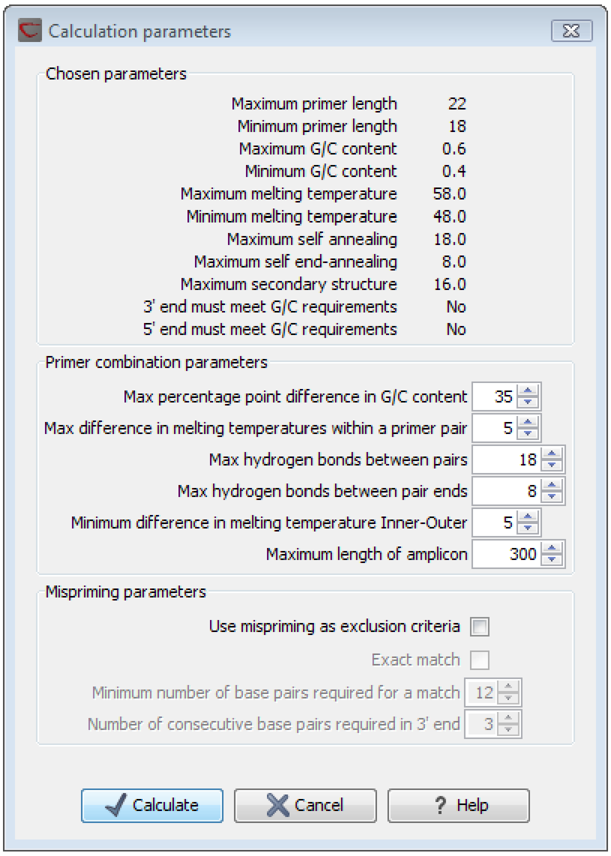
In *TaqMan* the user must thus define three regions: a *Forward primer region*, a *Reverse primer region*, and a *TaqMan probe region*. The easiest way to do this is to designate a *TaqMan primer/probe region* spanning the sequence region where TaqMan amplification is desired. This will automatically add all three regions to the sequence. If more control is desired about the placing of primers and probes the *Forward primer region*, *Reverse primer region* and *TaqMan probe region* can all be defined manually. If areas are known where primers or probes must not bind (e.g. repeat rich areas), one or more *No primers here* regions can be defined. The regions are defined by making a selection on the sequence and right-clicking the selection.

It is required that at least a part of the *Forward primer region* is located upstream of the *TaqMan Probe region*, and that the *TaqMan Probe region*, is located upstream of a part of the *Reverse primer region*.

In *TaqMan* mode the *Inner melting temperature* menu in the primer parameters panel is activated allowing the user to set a separate melting temperature interval for the TaqMan probe.

After exploring the available primers (see section 17.3) and setting the desired parameter values in the Primer Parameters preference group, the **Calculate** button will activate the primer design algorithm.

After pressing the **Calculate** button a dialog will appear (see figure 17.10) which is similar to the *Nested PCR* dialog described above (see section 17.6).



Calculation parameters

Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Primer combination parameters

Max percentage point difference in G/C content	35
Max difference in melting temperatures within a primer pair	5
Max hydrogen bonds between pairs	18
Max hydrogen bonds between pair ends	8
Minimum difference in melting temperature Inner-Outer	5
Maximum length of amplicon	300

Mispriming parameters

Use mispriming as exclusion criteria	☐
Exact match	☐
Minimum number of base pairs required for a match	12
Number of consecutive base pairs required in 3' end	3

Figure 17.10: *Calculation dialog*

In this dialog the options to set a minimum and a desired melting temperature difference between outer and inner refers to primer pair and probe respectively.

Furthermore, the central part of the dialog contains an additional parameter

- Maximum length of amplicon - determines the maximum length of the PCR fragment generated in the TaqMan analysis.

17.7.1 TaqMan output table

In TaqMan mode there are two primers and a probe in a given solution, forward primer (F), reverse primer (R) and a TaqMan probe (TP).

The output table can show primer/probe-pair combination parameters for all three combinations of primers and single primer parameters for both primers and the TaqMan probe (see section on Standard PCR for an explanation of the available primer-pair and single primer information).

The fragment length in this mode refers to the length of the PCR fragment generated by the primer pair, and this is also the PCR fragment which can be exported.

17.8 Sequencing primers

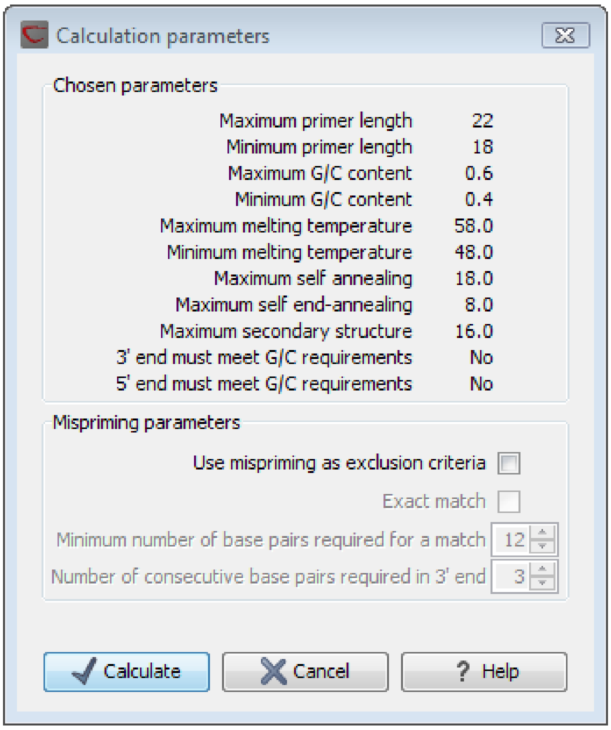
This mode is used to design primers for DNA sequencing.

In this mode the user can define a number of *Forward primer regions* and *Reverse primer regions* where a sequencing primer can start. These are defined by making a selection on the sequence and right-clicking the selection. If areas are known where primers must not bind (e.g. repeat rich areas), one or more *No primers here* regions can be defined.

No requirements are instated on the relative position of the regions defined.

After exploring the available primers (see section 17.3) and setting the desired parameter values in the Primer Parameters preference group, the **Calculate** button will activate the primer design algorithm.

After pressing the **Calculate** button a dialog will appear (see figure 17.11).



Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Mispriming parameters

Use mispriming as exclusion criteria □

Exact match □

Minimum number of base pairs required for a match 12

Number of consecutive base pairs required in 3' end 3

Figure 17.11: *Calculation dialog for sequencing primers*

Since design of sequencing primers does not require the consideration of interactions between primer pairs, this dialog is identical to the dialog shown in *Standard PCR* mode when only a single primer region is chosen. See the section 17.5 for a description.

17.8.1 Sequencing primers output table

In this mode primers are predicted independently for each region, but the optimal solutions are all presented in one table. The solutions are numbered consecutively according to their position on the sequence such that the forward primer region closest to the 5' end of the molecule is designated F1, the next one F2 etc.

For each solution, the single primer information described under Standard PCR is available in the table.

17.9 Alignment-based primer and probe design

CLC Genomics Workbench allows the user to design PCR primers and TaqMan probes based on an alignment of multiple sequences.

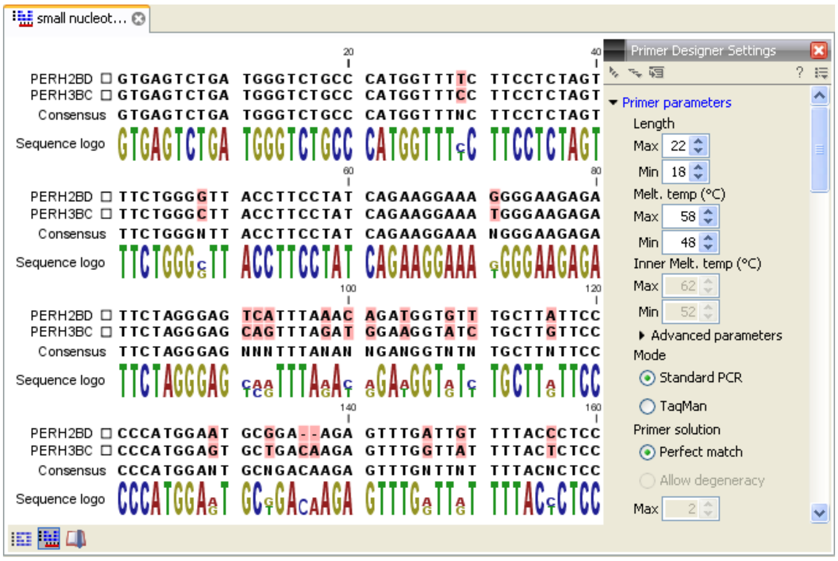
The primer designer for alignments can be accessed in two ways:

select alignment | Toolbox | Primers and Probes ( ) | Design Primers ( ) | OK

or **If the alignment is already open: | Click Primer Designer ( ) at the lower left part of the view**

In the alignment primer view (see figure 17.12), the basic options for viewing the template alignment are the same as for the standard view of alignments. See section 22 for an explanation of these options.

Note! This means that annotations such as e.g. known SNP's or exons can be displayed on the template sequence to guide the choice of primer regions. Since the definition of groups of sequences is essential to the primer design the selection boxes of the standard view are shown as default in the alignment primer view.



```
                      20                                  40
PERH2BD  □ GTGAGTCTGA TGGGTCTGCC CATGGTTTTC TTCCTCTAGT
PERH3BC  □ GTGAGTCTGA TGGGTCTGCC CATGGTTTCC TTCCTCTAGT
Consensus  GTGAGTCTGA TGGGTCTGCC CATGGTTTNC TTCCTCTAGT
Sequence logo
                      60                                  80
PERH2BD  □ TTCTGGGGTT ACCTTCCTAT CAGAAGGAAA GGGGAAGAGA
PERH3BC  □ TTCTGGGCTT ACCTTCCTAT CAGAAGGAAA TGGGAAGAGA
Consensus  TTCTGGGNTT ACCTTCCTAT CAGAAGGAAA NGGGAAGAGA
Sequence logo
                      100                                 120
PERH2BD  □ TTCTAGGGAG TCATTTAAAC AGATGGTGTT TGCTTATTCC
PERH3BC  □ TTCTAGGGAG CAGTTTAGAT GGAAGGTATC TGCTTGTTCC
Consensus  TTCTAGGGAG NNNTTTANAN NGANGGTNTN TGCTTNTTCC
Sequence logo
                      140                                 160
PERH2BD  □ CCCATGGAAT GCGGA--AGA GTTTGATTGT TTTACCCTCC
PERH3BC  □ CCCATGGAGT GCTGACAAGA GTTTGGTTAT TTTACTCTCC
Consensus  CCCATGGANT GCNGACAAGA GTTTGNTTNT TTTACNCTCC
Sequence logo
```

Primer Designer Settings

Primer parameters

Length
Max 22
Min 18

Melt. temp (°C)
Max 58
Min 48

Inner Melt. temp (°C)
Max 62
Min 52

Advanced parameters

Mode
● Standard PCR
○ TaqMan

Primer solution
● Perfect match
○ Allow degeneracy
Max 2

Figure 17.12: *The initial view of an alignment used for primer design.*

17.9.1 Specific options for alignment-based primer and probe design

Compared to the primer view of a single sequence the most notable difference is that the alignment primer view has no available graphical information. Furthermore, the selection boxes found to the right of the names in the alignment play an important role in specifying the oligo design process. This is elaborated below. The **Primer Parameters** group in the **Side Panel** has the same options for specifying primer requirements, but differs by the following (see figure 17.12):

- In the **Mode** submenu which specifies the reaction types the following options are found:
  - **Standard PCR.** Used when the objective is to design primers, or primer pairs, for PCR amplification of a single DNA fragment.
  - **TaqMan.** Used when the objective is to design a primer pair and a probe set for TaqMan quantitative PCR.
- The **Primer solution** submenu is used to specify requirements for the match of a PCR primer against the template sequences. These options are described further below. It contains the following options:
  - **Perfect match.**
  - **Allow degeneracy.**
  - **Allow mismatches.**

The work flow when designing alignment based primers and probes is as follows:

- Use selection boxes to specify groups of included and excluded sequences. To select all the sequences in the alignment, right-click one of the selection boxes and choose **Mark All**.
- Mark either a single forward primer region, a single reverse primer region or both on the sequence (and perhaps also a TaqMan region). Selections must cover all sequences in the included group. You can also specify that there should be no primers in a region (No Primers Here) or that a whole region should be amplified (Region to Amplify).
- Adjust parameters regarding single primers in the preference panel.
- Click the **Calculate** button.

17.9.2 Alignment based design of PCR primers

In this mode, a single or a pair of PCR primers are designed. *CLC Genomics Workbench* allows the user to design primers which will specifically amplify a group of *included* sequences but **not** amplify the remainder of the sequences, the *excluded* sequences. The selection boxes are used to indicate the status of a sequence, if the box is checked the sequence belongs to the included sequences, if not, it belongs to the excluded sequences. To design primers that are general for all primers in an alignment, simply add them all to the set of included sequences by checking all selection boxes. Specificity of priming is determined by criteria set by the user in the dialog box which is shown when the **Calculate** button is pressed (see below).

Different options can be chosen concerning the match of the primer to the template sequences in the included group:

- **Perfect match.** Specifies that the designed primers must have a perfect match to all relevant sequences in the alignment. When selected, primers will thus only be located in regions that are completely conserved within the sequences belonging to the included group.

- **Allow degeneracy.** Designs primers that may include ambiguity characters where heterogeneities occur in the included template sequences. The allowed fold of degeneracy is user defined and corresponds to the number of possible primer combinations formed by a degenerate primer. Thus, if a primer covers two 4-fold degenerate site and one 2-fold degenerate site the total fold of degeneracy is $4 * 4 * 2 = 32$ and the primer will, when supplied from the manufacturer, consist of a mixture of 32 different oligonucleotides. When scoring the available primers, degenerate primers are given a score which decreases with the fold of degeneracy.

- **Allow mismatches.** Designs primers which are allowed a specified number of mismatches to the included template sequences. The melting temperature algorithm employed includes the latest thermodynamic parameters for calculating T_m when single-base mismatches occur.

When in Standard PCR mode, clicking the **Calculate** button will prompt the dialog shown in figure 17.13.

The top part of this dialog shows the single-primer parameter settings chosen in the Primer parameters preference group which will be used by the design algorithm.

The central part of the dialog contains parameters pertaining to primer specificity (this is omitted if all sequences belong to the included group). Here, three parameters can be set:

- Minimum number of mismatches - the minimum number of mismatches that a primer must have against all sequences in the excluded group to ensure that it does not prime these.

- Minimum number of mismatches in 3' end - the minimum number of mismatches that a primer must have in its 3' end against all sequences in the excluded group to ensure that it does not prime these.

- Length of 3' end - the number of consecutive nucleotides to consider for mismatches in the 3' end of the primer.

The lower part of the dialog contains parameters pertaining to primer pairs (this is omitted when only designing a single primer). Here, three parameters can be set:

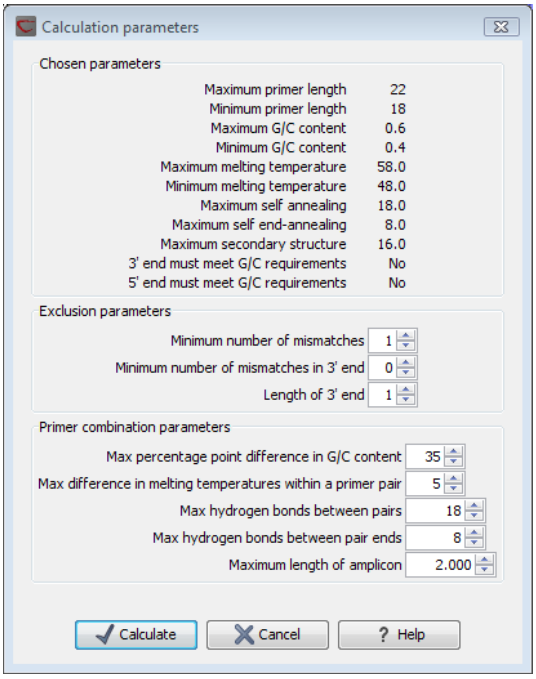
- Maximum percentage point difference in G/C content - if this is set at e.g. 5 points a pair of primers with 45% and 49% G/C nucleotides, respectively, will be allowed, whereas a pair of primers with 45% and 51% G/C nucleotides, respectively will not be included.

- Maximal difference in melting temperature of primers in a pair - the number of degrees Celsius that primers in a pair are all allowed to differ.

- Max hydrogen bonds between pairs - the maximum number of hydrogen bonds allowed between the forward and the reverse primer in a primer pair.

- Maximum length of amplicon - determines the maximum length of the PCR fragment.

The output of the design process is a table of single primers or primer pairs as described for primer design based on single sequences. These primers are specific to the included sequences in the alignment according to the criteria defined for specificity. The only novelty in the table, is that melting temperatures are displayed with both a maximum, a minimum and an average value to reflect that degenerate primers or primers with mismatches may have heterogeneous behavior on the different templates in the group of included sequences.



Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Exclusion parameters

Minimum number of mismatches	1
Minimum number of mismatches in 3' end	0
Length of 3' end	1

Primer combination parameters

Max percentage point difference in G/C content	35
Max difference in melting temperatures within a primer pair	5
Max hydrogen bonds between pairs	18
Max hydrogen bonds between pair ends	8
Maximum length of amplicon	2.000

Figure 17.13: *Calculation dialog shown when designing alignment based PCR primers.*

17.9.3 Alignment-based TaqMan probe design

CLC Genomics Workbench allows the user to design solutions for TaqMan quantitative PCR which consist of four oligos: a general primer pair which will amplify all sequences in the alignment, a specific TaqMan probe which will match the group of *included* sequences but **not** match the *excluded* sequences and a specific TaqMan probe which will match the group of *excluded* sequences but **not** match the *included* sequences. As above, the selection boxes are used to indicate the status of a sequence, if the box is checked the sequence belongs to the included sequences, if not, it belongs to the excluded sequences. We use the terms included and excluded here to be consistent with the section above although a probe solution is presented for both groups. In TaqMan mode, primers are not allowed degeneracy or mismatches to any template sequence in the alignment, variation is only allowed/required in the TaqMan probes.

Pushing the **Calculate** button will cause the dialog shown in figure 17.14 to appear.

The top part of this dialog is identical to the *Standard PCR* dialog for designing primer pairs described above.

The central part of the dialog contains parameters to define the specificity of TaqMan probes. Two parameters can be set:

- Minimum number of mismatches - the minimum total number of mismatches that must exist between a specific TaqMan probe and all sequences which belong to the group not recognized by the probe.

- Minimum number of mismatches in central part - the minimum number of mismatches in the central part of the oligo that must exist between a specific TaqMan probe and all sequences which belong to the group not recognized by the probe.

The lower part of the dialog contains parameters pertaining to primer pairs and the comparison between the outer oligos(primers) and the inner oligos (TaqMan probes). Here, five options can be set:

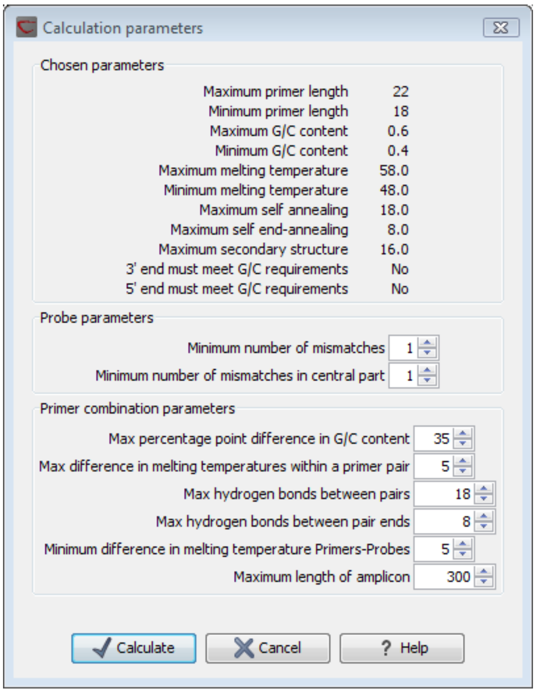
- Maximum percentage point difference in G/C content (described above under Standard PCR).
- Maximal difference in melting temperature of primers in a pair - the number of degrees Celsius that primers in the primer pair are all allowed to differ.
- Maximum pair annealing score - the maximum number of hydrogen bonds allowed between the forward and the reverse primer in an oligo pair. This criteria is applied to all possible combinations of primers and probes.
- Minimum difference in the melting temperature of primer (outer) and TaqMan probe (inner) oligos - all comparisons between the melting temperature of primers and probes must be at least this different, otherwise the solution set is excluded.
- Desired temperature difference in melting temperature between outer (primers) and inner (TaqMan) oligos - the scoring function discounts solution sets which deviate greatly from this value. Regarding this, and the minimum difference option mentioned above, please note that to ensure flexibility there is no directionality indicated when setting parameters for melting temperature differences between probes and primers, i.e. it is not specified whether the probes should have a lower or higher T_m. Instead this is determined by the allowed temperature intervals for inner and outer oligos that are set in the primer parameters preference group in the side panel. If a higher T_m of probes is required, choose a T_m interval for probes which has higher values than the interval for outer primers.

The output of the design process is a table of solution sets. Each solution set contains the following: a set of primers which are general to all sequences in the alignment, a TaqMan probe which is specific to the set of included sequences (sequences where selection boxes are checked) and a TaqMan probe which is specific to the set of excluded sequences (marked by *). Otherwise, the table is similar to that described above for TaqMan probe prediction on single sequences.

17.10 Analyze primer properties

CLC Genomics Workbench can calculate and display the properties of predefined primers and probes:

> **select a primer sequence (primers are represented as DNA sequences in the Navigation Area) | Toolbox in the Menu Bar | Primers and Probes ( ) | Analyze Primer Properties ( )**



Chosen parameters

Maximum primer length	22
Minimum primer length	18
Maximum G/C content	0.6
Minimum G/C content	0.4
Maximum melting temperature	58.0
Minimum melting temperature	48.0
Maximum self annealing	18.0
Maximum self end-annealing	8.0
Maximum secondary structure	16.0
3' end must meet G/C requirements	No
5' end must meet G/C requirements	No

Probe parameters

Minimum number of mismatches	1
Minimum number of mismatches in central part	1

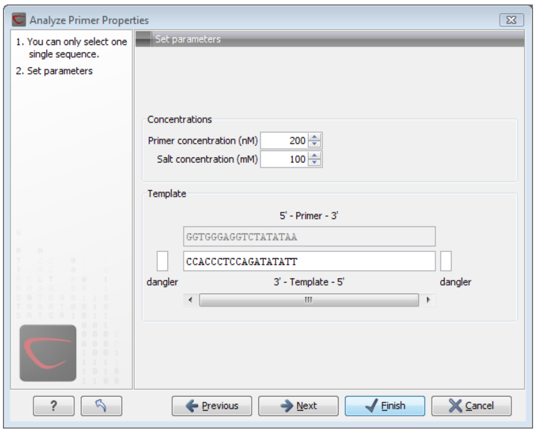
Primer combination parameters

Max percentage point difference in G/C content	35
Max difference in melting temperatures within a primer pair	5
Max hydrogen bonds between pairs	18
Max hydrogen bonds between pair ends	8
Minimum difference in melting temperature Primers-Probes	5
Maximum length of amplicon	300

Figure 17.14: *Calculation dialog shown when designing alignment based TaqMan probes.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove a sequence from the selected elements.

Clicking **Next** generates the dialog seen in figure 17.15:



1. You can only select one single sequence.
2. Set parameters

Concentrations

Primer concentration (nM) 200

Salt concentration (mM) 100

Template

5' - Primer - 3'

GGTGGGAGGTCTATATAA

CCACCCTCCAGATATATT

dangler 3' - Template - 5' dangler

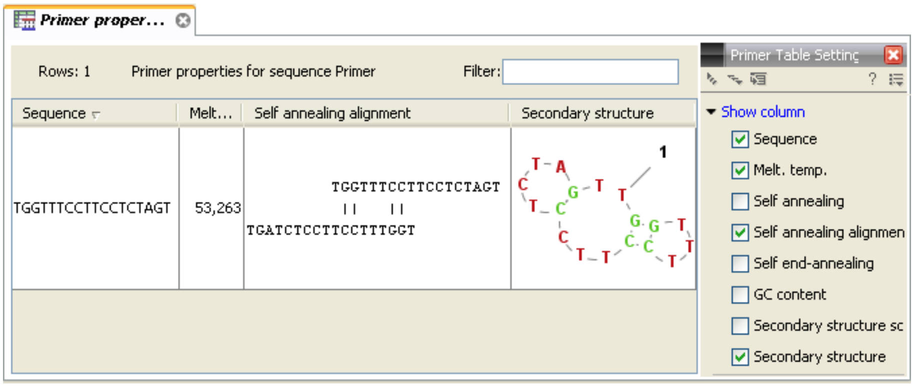
Figure 17.15: *The parameters for analyzing primer properties.*

In the *Concentrations* panel a number of parameters can be specified concerning the reaction mixture and which influence melting temperatures

- **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM)
- **Salt concentration.** Specifies the concentration of monovalent cations ($[NA^+]$, $[K^+]$ and equivalents) in units of millimoles (mM)

In the *Template panel* the sequences of the chosen primer and the template sequence are shown. The template sequence is as default set to the reverse complement of the primer sequence i.e. as perfectly base-pairing. However, it is possible to edit the template to introduce mismatches which may affect the melting temperature. At each side of the template sequence a text field is shown. Here, the dangling ends of the template sequence can be specified. These may have an important affect on the melting temperature [Bommarito et al., 2000]

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The result is shown in figure 17.16:



Primer proper...

Rows: 1 Primer properties for sequence Primer Filter:

Sequence	Melt...	Self annealing alignment	Secondary structure
TGGTTTCCTTCCTCTAGT	53,263	TGGTTTCCTTCCTCTAGT || || TGATCTCCTTCCTTTGGT	

Primer Table Setting

▾ Show column

- ☑ Sequence
- ☑ Melt. temp.
- □ Self annealing
- ☑ Self annealing alignmen
- □ Self end-annealing
- □ GC content
- □ Secondary structure sc
- ☑ Secondary structure

Figure 17.16: *Properties of a primer from the Example Data.*

In the **Side Panel** you can specify the information to display about the primer. The information parameters of the primer properties table are explained in section 17.5.2.

17.11 Find binding sites and create fragments

In *CLC Genomics Workbench* you have the possibility of matching known primers against one or more DNA sequences or a list of DNA sequences. This can be applied to test whether a primer used in a previous experiment is applicable to amplify e.g. a homologous region in another species, or to test for potential mispriming. This functionality can also be used to extract the resulting PCR product when two primers are matched. This is particularly useful if your primers have extensions in the 5' end.

To search for primer binding sites:

Toolbox | **Primers and Probes (** **)** | **Find Binding Sites and Create Fragments (** **)**

If a sequence was already selected, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

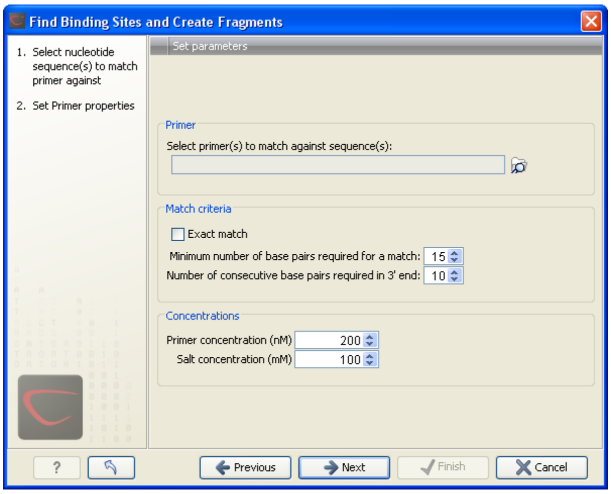
Click **Next** when all the sequence have been added.

Note! You should not add the primer sequences at this step.

17.11.1 Binding parameters

This opens the dialog displayed in figure 17.17:

At the top, select one or more primers by clicking the browse ( ) button. In *CLC Genomics*



1. Select nucleotide sequence(s) to match primer against
2. Set Primer properties

Primer

Select primer(s) to match against sequence(s):

Match criteria

☐ Exact match

Minimum number of base pairs required for a match: 15

Number of consecutive base pairs required in 3' end: 10

Concentrations

Primer concentration (nM) 200

Salt concentration (mM) 100

Figure 17.17: *Search parameters for finding primer binding sites.*

Workbench, primers are just DNA sequences like any other, but there is a filter on the length of the sequence. Only sequences up to 400 bp can be added.

The **Match criteria** for matching a primer to a sequence are:

- **Exact match**. Choose only to consider exact matches of the primer, i.e. all positions must base pair with the template.
- **Minimum number of base pairs required for a match**. How many nucleotides of the primer that must base pair to the sequence in order to cause priming/mispriming.
- **Number of consecutive base pairs required in 3' end**. How many consecutive 3' end base pairs in the primer that MUST be present for priming/mispriming to occur. This option is included since 3' terminal base pairs are known to be essential for priming to occur.

Note that the number of mismatches is reported in the output, so you will be able to filter on this afterwards (see below).

Below the match settings, you can adjust **Concentrations** concerning the reaction mixture. This is used when reporting melting temperatures for the primers.

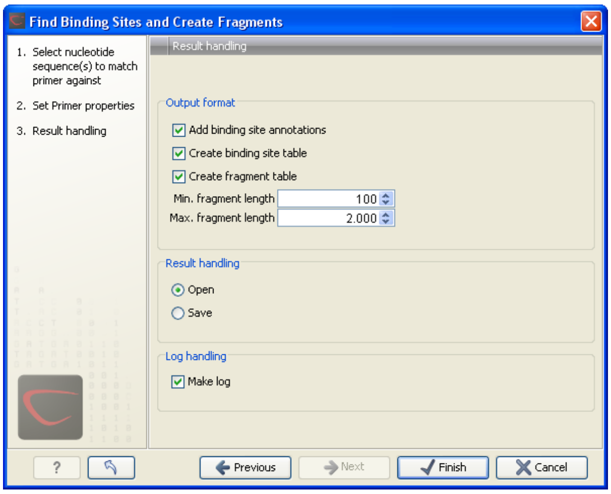
- **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM)
- **Salt concentration.** Specifies the concentration of monovalent cations ($[NA^+]$, $[K^+]$ and equivalents) in units of millimoles (mM)

17.11.2 Results - binding sites and fragments

Click **Next** to specify the output options as shown in figure 17.18:

The output options are:

- **Add binding site annotations**. This will add annotations to the input sequences (see details below).



Find Binding Sites and Create Fragments

1. Select nucleotide sequence(s) to match primer against
2. Set Primer properties
3. Result handling

Result handling

Output format

☑ Add binding site annotations

☑ Create binding site table

☑ Create fragment table

Min. fragment length 100

Max. fragment length 2.000

Result handling

● Open

○ Save

Log handling

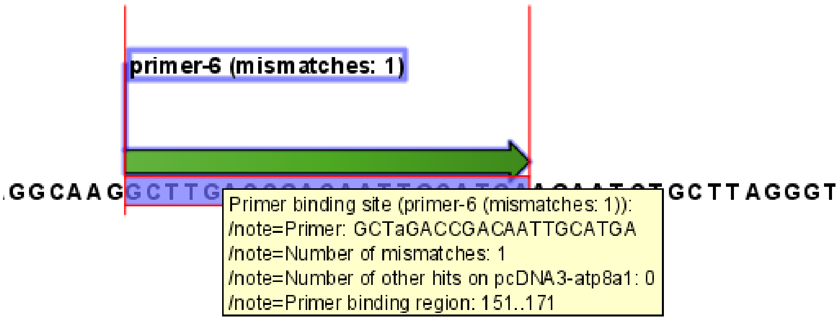
☑ Make log

Figure 17.18: *Output options include reporting of binding sites and fragments.*

- **Create binding site table**. Creates a table of all binding sites. Described in details below.
- **Create fragment table**. Showing a table of all fragments that could result from using the primers. Note that you can set the minimum and maximum sizes of the fragments to be shown. The table is described in detail below.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

An example of a **binding site annotation** is shown in figure 17.19.

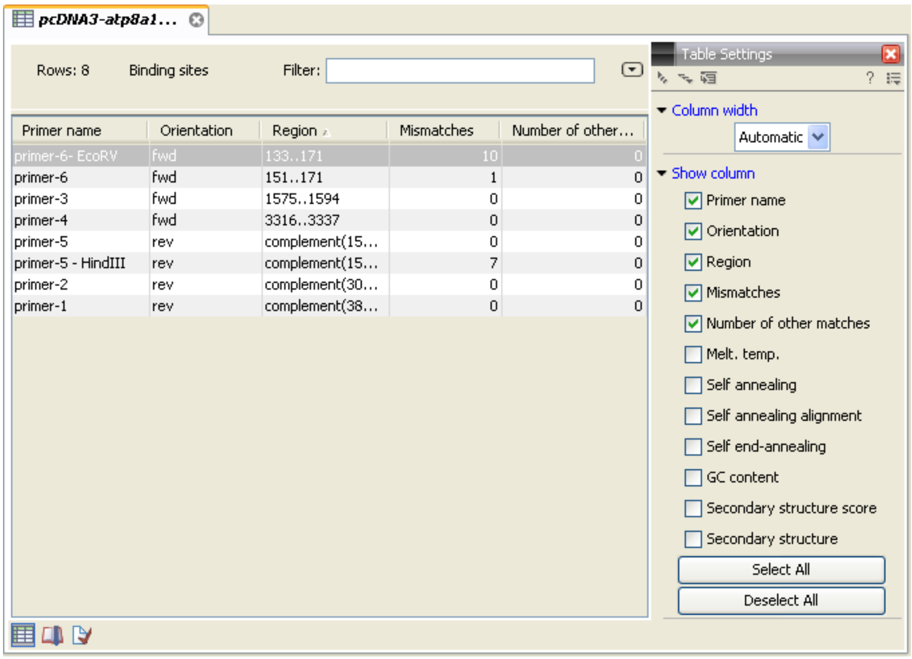


Primer binding site (primer-6 (mismatches: 1)):
/note=Primer: GCTaGACCGACAATTGCATGA
/note=Number of mismatches: 1
/note=Number of other hits on pcDNA3-atp8a1: 0
/note=Primer binding region: 151..171

Figure 17.19: *Annotation showing a primer match.*

The annotation has the following information:

- **Sequence of the primer.** Positions with mismatches will be in lower-case (see the fourth position in figure 17.19 where the primer has an a and the template sequence has a T).
- **Number of mismatches.**
- **Number of other hits on the same sequence**. This number can be useful to check specificity of the primer.
- **Binding region**. This region ends with the 3' exact match and is simply the primer length upstream. This means that if you have 5' extensions to the primer, part of the binding region covers sequence that will actually not be annealed to the primer.



Rows: 8 Binding sites Filter:

Primer name	Orientation	Region	Mismatches	Number of other...
primer-6- EcoRV	fwd	133..171	10	0
primer-6	fwd	151..171	1	0
primer-3	fwd	1575..1594	0	0
primer-4	fwd	3316..3337	0	0
primer-5	rev	complement(15...	0	0
primer-5 - HindIII	rev	complement(15...	7	0
primer-2	rev	complement(30...	0	0
primer-1	rev	complement(38...	0	0

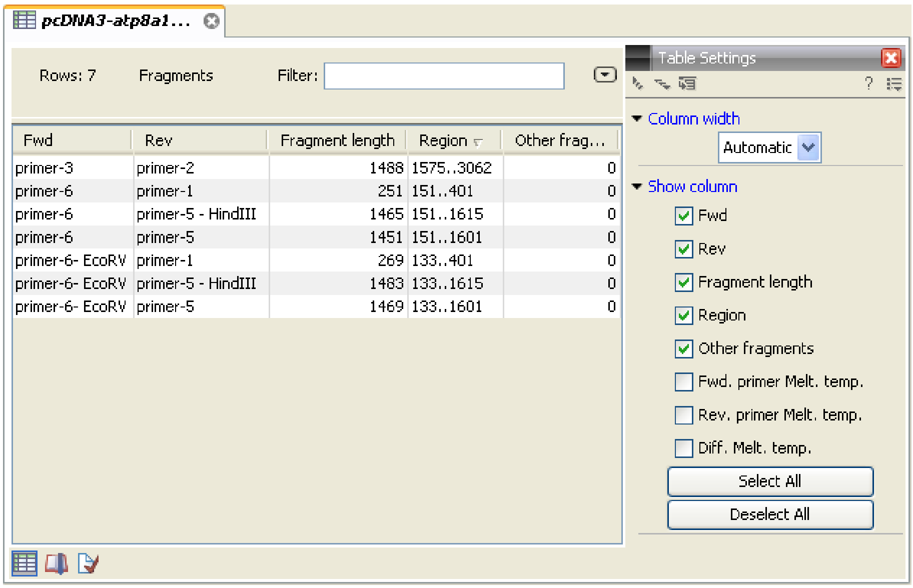
Show column
☑ Primer name
☑ Orientation
☑ Region
☑ Mismatches
☑ Number of other matches
□ Melt. temp.
□ Self annealing
□ Self annealing alignment
□ Self end-annealing
□ GC content
□ Secondary structure score
□ Secondary structure

Figure 17.20: *A table showing all binding sites.*

An example of the **primer binding site table** is shown in figure 17.20.

The information here is the same as in the primer annotation and furthermore you can see additional information about melting temperature etc. by selecting the options in the **Side Panel**. See a more detailed description of this information in section 17.5.2. You can use this table to browse the binding sites. If you make a split view of the table and the sequence (see section 3.2.6), you can browse through the binding positions by clicking in the table. This will cause the sequence view to jump to the position of the binding site.

An example of a **fragment table** is shown in figure 17.21.



Rows: 7 Fragments Filter:

Fwd	Rev	Fragment length	Region	Other frag...
primer-3	primer-2	1488	1575..3062	0
primer-6	primer-1	251	151..401	0
primer-6	primer-5 - HindIII	1465	151..1615	0
primer-6	primer-5	1451	151..1601	0
primer-6- EcoRV	primer-1	269	133..401	0
primer-6- EcoRV	primer-5 - HindIII	1483	133..1615	0
primer-6- EcoRV	primer-5	1469	133..1601	0

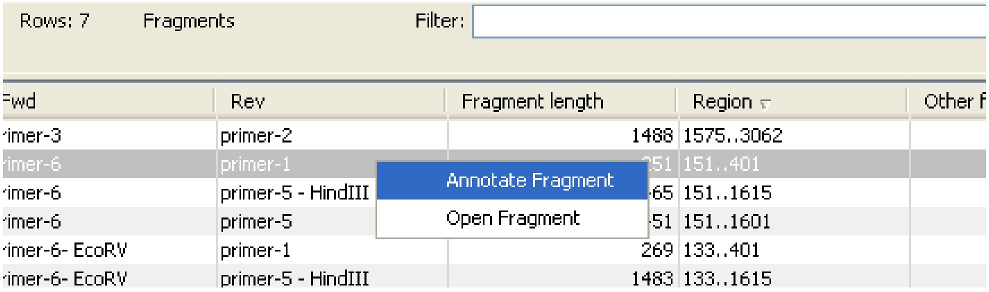
Show column
☑ Fwd
☑ Rev
☑ Fragment length
☑ Region
☑ Other fragments
□ Fwd. primer Melt. temp.
□ Rev. primer Melt. temp.
□ Diff. Melt. temp.

Figure 17.21: *A table showing all possible fragments of the specified size.*

The table first lists the names of the forward and reverse primers, then the length of the fragment and the region. The last column tells if there are other possible fragments fulfilling the length criteria on this sequence. This information can be used to check for competing products in the PCR. In the **Side Panel** you can show information about melting temperature for the primers as well as the difference between melting temperatures.

You can use this table to browse the fragment regions. If you make a split view of the table and the sequence (see section 3.2.6), you can browse through the fragment regions by clicking in the table. This will cause the sequence view to jump to the start position of the fragment.

There are some additional options in the fragment table. First, you can annotate the fragment on the original sequence. This is done by right-clicking (Ctrl-click on Mac) the fragment and choose **Annotate Fragment** as shown in figure 17.22.



Rows: 7 Fragments Filter:

wd	Rev	Fragment length	Region	Other f
imer-3	primer-2	1488	1575..3062	
imer-6	primer-1	51	151..401	
imer-6	primer-5 - HindIII	65	151..1615	
imer-6	primer-5	51	151..1601	
imer-6- EcoRV	primer-1	269	133..401	
imer-6- EcoRV	primer-5 - HindIII	1483	133..1615	

Figure 17.22: *Right-clicking a fragment allows you to annotate the region on the input sequence or open the fragment as a new sequence.*

This will put a *PCR fragment* annotations on the input sequence covering the region specified in the table. As you can see from figure 17.22, you can also choose to **Open Fragment**. This will create a new sequence representing the PCR product that would be the result of using these two primers. Note that if you have extensions on the primers, they will be used to construct the new sequence. If you are doing restriction cloning using primers with restriction site extensions, you can use this functionality to retrieve the PCR fragment for us in the cloning editor (see section 21.1).

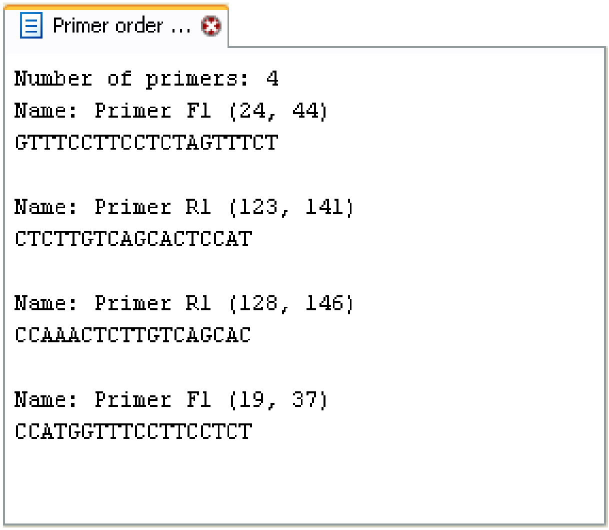
17.12 Order primers

To facilitate the ordering of primers and probes, *CLC Genomics Workbench* offers an easy way of displaying, and saving, a textual representation of one or more primers:

> **select primers in Navigation Area | Toolbox in the Menu Bar | Primers and Probes ( ) | Order Primers ( )**

This opens a dialog where you can choose additional primers. Clicking **OK** opens a textual representation of the primers (see figure 17.23). The first line states the number of primers being ordered and after this follows the names and nucleotide sequences of the primers in 5'-3' orientation. From the editor, the primer information can be copied and pasted to web forms or e-mails. The created object can also be saved and exported as a text file.

See figure 17.23



```
Number of primers: 4
Name: Primer F1 (24, 44)
GTTTCCTTCCTCTAGTTTCT

Name: Primer R1 (123, 141)
CTCTTGTCAGCACTCCAT

Name: Primer R1 (128, 146)
CCAAACTCTTGTCAGCAC

Name: Primer F1 (19, 37)
CCATGGTTTCCTTCCTCT
```

Figure 17.23: *A primer order for 4 primers.*

Chapter 18

Sequencing data analyses

Contents

CLC Genomics Workbench lets you import, trim and assemble DNA sequence reads from automated sequencing machines. A number of different formats are supported (see section 7.1.1). This chapter first explains how to trim sequence reads. Next follows a description of how to assemble reads into contigs both with and without a reference sequence. In the final section, the options for viewing and editing contigs are explained.

18.1 Importing and viewing trace data

A number of different binary trace data formats can be imported into the program, including *Standard Chromatogram Format (.SCF)*, *ABI sequencer data files (.ABI and .AB1)*, *PHRED output files (.PHD)* and *PHRAP output files (.ACE)* (see section 7.1.1).

After import, the sequence reads and their trace data are saved as DNA sequences. This means that all analyzes which apply to DNA sequences can be performed on the sequence reads, including e.g. BLAST and open reading frame prediction.

You can see additional information about the quality of the traces by holding the mouse cursor on the imported sequence. This will display a tool tip as shown in figure 18.1.

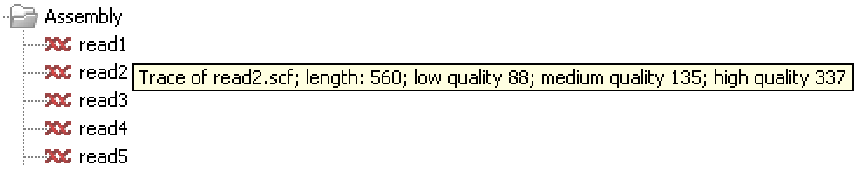

Assembly
read1
read2 Trace of read2.scf; length: 560; low quality 88; medium quality 135; high quality 337
read3
read4
read5

Figure 18.1: *A tooltip displaying information about the quality of the chromatogram.*

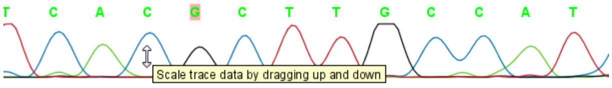
If the trace file does not contain information about quality, only the sequence length will be shown.

To view the trace data, open the sequence read in a standard sequence view ().

18.1.1 Scaling traces

The traces can be scaled by dragging the trace vertically as shown in figure figure 18.2. The Workbench automatically adjust the height of the traces to be readable, but if the trace height varies a lot, this manual scaling is very useful.

The height of the area available for showing traces can be adjusted in the **Side Panel** as described insection 18.1.2.



Scale trace data by dragging up and down

Figure 18.2: *Grab the traces to scale.*

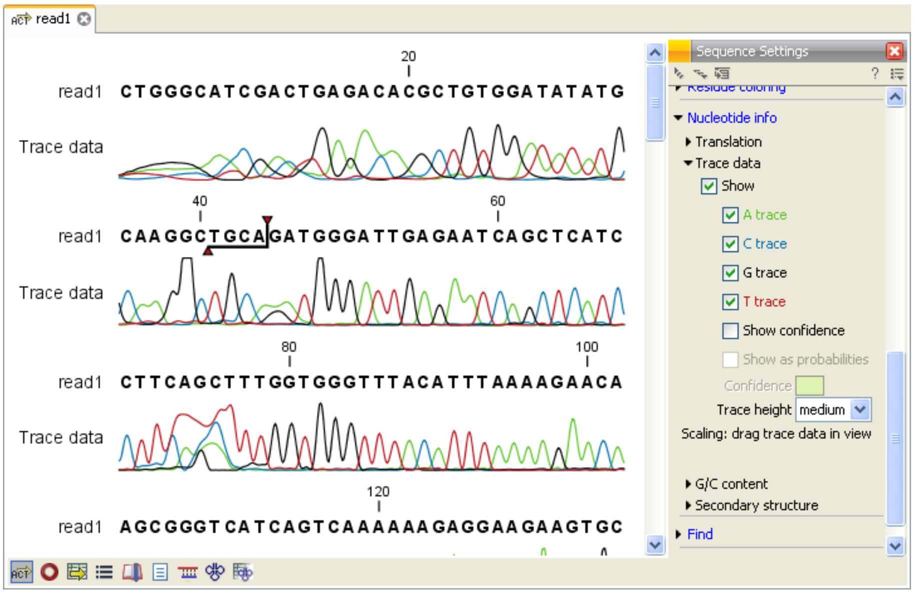
18.1.2 Trace settings in the Side Panel

In the Nucleotide info preference group the display of trace data can be selected and unselected. When selected, the trace data information is shown as a plot beneath the sequence. The appearance of the plot can be adjusted using the following options (see figure 18.3):

- **Nucleotide trace.** For each of the four nucleotides the trace data can be selected and unselected.
- **Scale traces.** A slider which allows the user to scale the height of the trace area. Scaling the traces individually is described in section 18.1.1.

18.2 Trim sequences

CLC Genomics Workbench offers a number of ways to trim your sequence reads prior to assembly. Trimming can be done either as a separate task before assembling, or it can be performed as an


read1
20
read1 CTGGGCATCGACTGAGACACGCTGTGGATATATG
Trace data
40
60
read1 CAAGGCTGCAGATGGGATTGAGAATCAGCTCATC
Trace data
80
100
read1 CTTCAGCTTTGGTGGGTTTACATTTAAAAGAACA
Trace data
120
read1 AGCGGGTCATCAGTCAAAAAAGAGGAAGAAGTGC
Sequence Settings
Residue coloring

Nucleotide info
- Translation
- Trace data
  - ☑ Show
    - ☑ A trace
    - ☑ C trace
    - ☑ G trace
    - ☑ T trace
    - □ Show confidence
    - □ Show as probabilities
    - Confidence
    - Trace height: medium
  - Scaling: drag trace data in view
- G/C content
- Secondary structure

Find

Figure 18.3: *A sequence with trace data. The preferences for viewing the trace are shown in the Side Panel.*

integrated part of the assembly process (see section 18.3).

Trimming as a separate task can be done either manually or automatically.

In both instances, trimming of a sequence does not cause data to be deleted, instead both the manual and automatic trimming will put a "Trim" annotation on the trimmed parts as an indication to the assembly algorithm that this part of the data is to be ignored (see figure 18.4). This means that the effect of different trimming schemes can easily be explored without the loss of data. To remove existing trimming from a sequence, simply remove its trim annotation (see section 10.3.2).

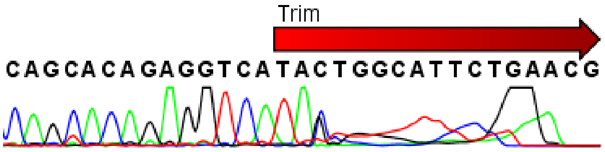


Figure 18.4: *Trimming creates annotations on the regions that will be ignored in the assembly process.*

18.2.1 Manual trimming

Sequence reads can be trimmed manually while inspecting their trace and quality data. Trimming sequences manually corresponds to adding annotation (see also section 10.3.2) but is special in the sense that trimming can only be applied to the ends of a sequence:

> **double-click the sequence to trim in the Navigation Area | select the region you want to trim | right-click the selection | Trim sequence left/right to determine the direction of the trimming**

This will add trimming annotation to the end of the sequence in the selected direction.

18.2.2 Automatic trimming

Sequence reads can be trimmed automatically based on a number of different criteria. Automatic trimming is particularly useful in the following situations:

- If you have many sequence reads to be trimmed.
- If you wish to trim vector contamination from sequence reads.
- If you wish to ensure that the trimming is done according to the same criteria for all the sequence reads.

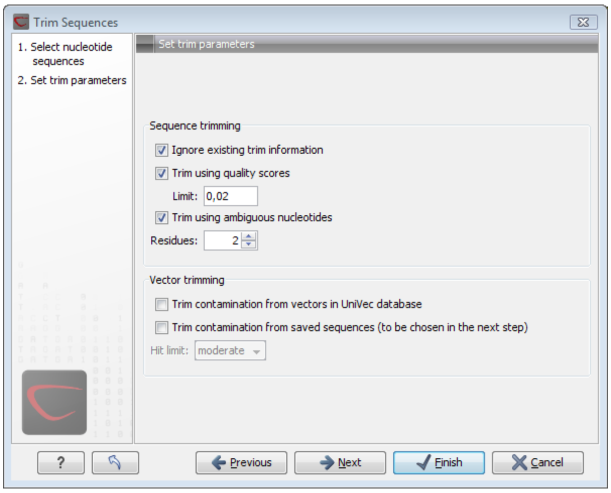
To trim sequences automatically:

> **select sequence(s) or sequence lists to trim** | **Toolbox in the Menu Bar** | **Sequencing Data Analyses** ( ) | **Trim Sequences** ( )

This opens a dialog where you can alter your choice of sequences.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 18.5.



Trim Sequences

1. Select nucleotide sequences
2. Set trim parameters

Set trim parameters

Sequence trimming

☑ Ignore existing trim information

☑ Trim using quality scores

Limit: 0,02

☑ Trim using ambiguous nucleotides

Residues: 2

Vector trimming

□ Trim contamination from vectors in UniVec database

□ Trim contamination from saved sequences (to be chosen in the next step)

Hit limit: moderate

Figure 18.5: *Setting parameters for trimming.*

The following parameters can be adjusted in the dialog:

- **Ignore existing trim information.** If you have previously trimmed the sequences, you can check this to remove existing trimming annotation prior to analysis.
- **Trim using quality scores.** If the sequence files contain quality scores from a base-caller algorithm this information can be used for trimming sequence ends. The program uses the modified-Mott trimming algorithm for this purpose (Richard Mott, personal communication):

  Quality scores in the Workbench are on a Phred scale in the Workbench (formats using other scales are converted during import). First step in the trim process is to convert the quality score (*Q*) to error probability: $p_{error} = 10^{\frac{Q}{-10}}$. (This now means that low values are high quality bases.)

Next, for every base a new value is calculated: $Limit - p_{error}$. This value will be negative for low quality bases, where the error probability is high.

For every base, the Workbench calculates the running sum of this value. If the sum drops below zero, it is set to zero. The part of the sequence to be retained after trimming is the region between the first positive value of the running sum and the highest value of the running sum. Everything before and after this region will be trimmed off.

A read will be completely removed if the score never makes it above zero.

At http://www.clcbio.com/files/usermanuals/trim.zip you find an example sequence and an Excel sheet showing the calculations done for this particular sequence to illustrate the procedure described above.

- **Trim ambiguous nucleotides.** This option trims the sequence ends based on the presence of ambiguous nucleotides (typically N). Note that the automated sequencer generating the data must be set to output ambiguous nucleotides in order for this option to apply. The algorithm takes as input the *maximal number of ambiguous nucleotides allowed in the sequence after trimming*. If this maximum is set to e.g. 3, the algorithm finds the maximum length region containing 3 or fewer ambiguities and then trims away the ends not included in this region.

- **Trim contamination from vectors in UniVec database.** If selected, the program will match the sequence reads against all vectors in the UniVec database and remove sequence ends with significant matches (the database is included when you install the *CLC Genomics Workbench*). A list of all the vectors in the UniVec database can be found at http://www.ncbi.nlm.nih.gov/VecScreen/replist.html.

  - **Hit limit.** Specifies how strictly vector contamination is trimmed. Since vector contamination usually occurs at the beginning or end of a sequence, different criteria are applied for terminal and internal matches. A match is considered terminal if it is located within the first 25 bases at either sequence end. Three match categories are defined according to the expected frequency of an alignment with the same score occurring between random sequences. The *CLC Genomics Workbench* uses the same settings as VecScreen (http://www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html):

    - **Weak.** Expect 1 random match in 40 queries of length 350 kb
      - Terminal match with Score 16 to 18.
      - Internal match with Score 23 to 24.
    - **Moderate.** Expect 1 random match in 1,000 queries of length 350 kb
      - Terminal match with Score 19 to 23.
      - Internal match with Score 25 to 29.
    - **Strong.** Expect 1 random match in 1,000,000 queries of length 350 kb
      - Terminal match with Score $\geq$ 24.
      - Internal match with Score $\geq$ 30.

    Note that selecting e.g. **Weak** will also include matches in the **Moderate** and **Strong** categories.

- **Trim contamination from saved sequences.** This option lets you select your own vector sequences that you know might be the cause of contamination. If you select this option, you will be able to select one or more sequences when you click **Next**.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will start the trimming process. Views of each trimmed sequence will be shown, and you can inspect the result by looking at the "Trim" annotations (they are colored red as default). If there are no trim annotations, the sequence has not been trimmed.

18.3 Assemble sequences

This section describes how to assemble a number of sequence reads into a contig without the use of a reference sequence (a known sequence that can be used for comparison with the other sequences, see section 18.4). To perform the assembly:

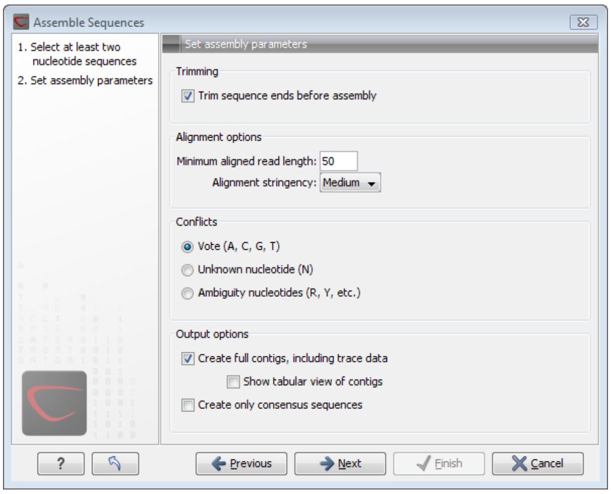
> **select sequences to assemble** | **Toolbox in the Menu Bar** | **Sequencing Data Analyses** (**) | **Assemble Sequences** (**)**

This opens a dialog where you can alter your choice of sequences which you want to assemble. You can also add sequence lists.

Note! You can assemble a maximum of 2000 sequences at a time.

To assemble more sequences, please use the **De Novo Assembly** () under **High-throughput Sequencing** (**)** in the **Toolbox**.

When the sequences are selected, click **Next**. This will show the dialog in figure 18.6



Assemble Sequences

1. Select at least two nucleotide sequences
2. Set assembly parameters

Set assembly parameters

Trimming

☑ Trim sequence ends before assembly

Alignment options

Minimum aligned read length: 50

Alignment stringency: Medium

Conflicts

● Vote (A, C, G, T)

○ Unknown nucleotide (N)

○ Ambiguity nucleotides (R, Y, etc.)

Output options

☑ Create full contigs, including trace data

☐ Show tabular view of contigs

☐ Create only consensus sequences

Figure 18.6: *Setting assembly parameters.*

This dialog gives you the following options for assembling:

- **Trim sequence ends before assembly.** If you have not previously trimmed the sequences, this can be done by checking this box. If selected, the next step in the dialog will allow you to specify settings for trimming (see section 19.3.1).
- **Minimum aligned read length.** The minimum number of nucleotides in a read which must be successfully aligned to the contig. If this criteria is not met by a read, the read is excluded from the assembly.
- **Alignment stringency.** Specifies the stringency of the scoring function used by the alignment step in the contig assembly algorithm. A higher stringency level will tend to produce contigs

with less ambiguities but will also tend to omit more sequencing reads and to generate more and shorter contigs. Three stringency levels can be set:

  - **Low.**
  - **Medium.**
  - **High.**

- **Conflicts.** If there is a conflict, i.e. a position where there is disagreement about the residue (A, C, T or G), you can specify how the contig sequence should reflect the conflict:

  - **Vote (A, C, G, T).** The conflict will be solved by counting instances of each nucleotide and then letting the majority decide the nucleotide in the contig. In case of equality, ACGT are given priority over one another in the stated order.
  - **Unknown nucleotide (N).** The contig will be assigned an 'N' character in all positions with conflicts.
  - **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the reads. For an overview of ambiguity codes, see Appendix I.

  Note, that conflicts will always be highlighted no matter which of the options you choose. Furthermore, each conflict will be marked as annotation on the contig sequence and will be present if the contig sequence is extracted for further analysis. As a result, the details of any experimental heterogeneity can be maintained and used when the result of single-sequence analyzes is interpreted. Read more about conflicts in section 18.6.4.

- **Create full contigs, including trace data.** This will create a contig where all the aligned reads are displayed below the contig sequence. (You can always extract the contig sequence without the reads later on.) For more information on how to use the contigs that are created, see section 18.6.

- **Show tabular view of contigs.** A contig can be shown both in a graphical as well as a tabular view. If you select this option, a tabular view of the contig will also be opened (Even if you do not select this option, you can show the tabular view of the contig later on by clicking **Table** (▦) at the bottom of the view.) For more information about the tabular view of contigs, see section 18.6.7.

- **Create only consensus sequences.** This will not display a contig but will only output the assembled contig sequences as single nucleotide sequences. If you choose this option it is not possible to validate the assembly process and edit the contig based on the traces.

If you have chosen to "Trim sequences", click **Next** and you will be able to set trim parameters (see section 19.3.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

When the assembly process has ended, a number of views will be shown, each containing a contig of two or more sequences that have been matched. If the number of contigs seem too high or low, try again with another **Alignment stringency** setting. Depending on your choices of output options above, the views will include trace files or only contig sequences. However, the calculation of the contig is carried out the same way, no matter how the contig is displayed.

See section 18.6 on how to use the resulting contigs.

18.4 Assemble to reference sequence

This section describes how to assemble a number of sequence reads into a contig using a reference sequence. A reference sequence can be particularly helpful when the objective is to characterize SNP variation in the data.

To start the assembly:

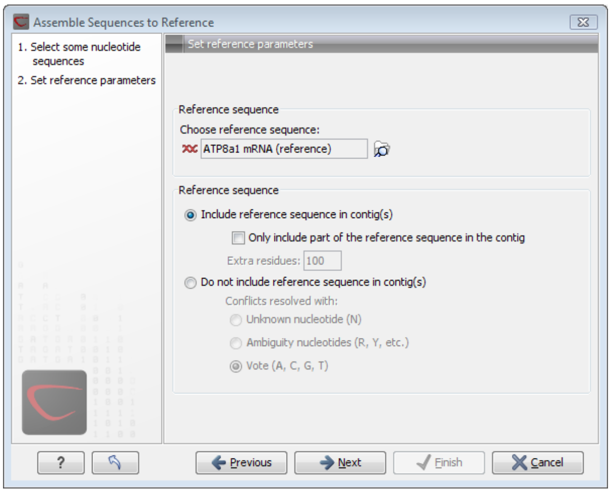
> **select sequences to assemble | Toolbox in the Menu Bar | Sequencing Data Analyses ( ) | Assemble Sequences to Reference ( )**

This opens a dialog where you can alter your choice of sequences which you want to assemble. You can also add sequence lists.

Note! You can assemble a maximum of 2000 sequences at a time.

To assemble more sequences, please use the **Map Reads to Reference** ( ) under **High-throughput Sequencing ( )** in the **Toolbox**.

When the sequences are selected, click **Next**, and you will see the dialog shown in figure 18.7



Assemble Sequences to Reference

1. Select some nucleotide sequences
2. Set reference parameters

Set reference parameters

Reference sequence

Choose reference sequence: ATP8a1 mRNA (reference)

Reference sequence

● Include reference sequence in contig(s)

□ Only include part of the reference sequence in the contig

Extra residues: 100

○ Do not include reference sequence in contig(s)

Conflicts resolved with:

○ Unknown nucleotide (N)

○ Ambiguity nucleotides (R, Y, etc.)

● Vote (A, C, G, T)

Figure 18.7: *Setting assembly parameters when assembling to a reference sequence.*

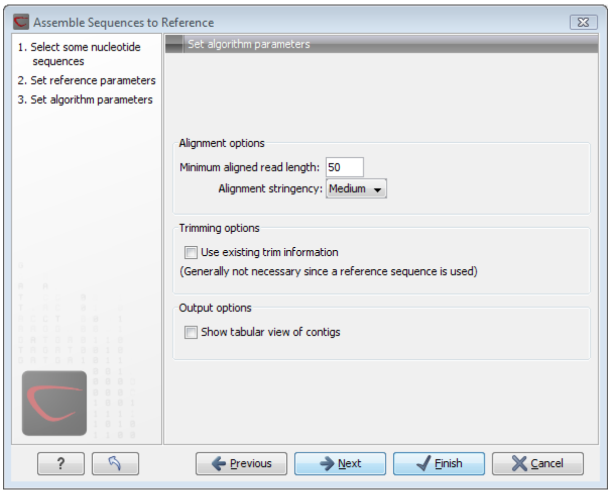
This dialog gives you the following options for assembling:

- **Reference sequence.** Click the **Browse and select element** icon ( ) in order to select a sequence to use as reference.
- **Include reference sequence in contig(s).** This will display a contig data-object with the reference sequence at the top and the reads aligned below. This option is useful when comparing sequence reads to a closely related reference sequence e.g. when sequencing for SNP characterization.
  - **Only include part of the reference sequence in the contig.** If the aligned sequence reads only cover a small part of the reference sequence, it may not be desirable to include the whole reference sequence in the contig data-object. When selected, this option lets you specify how many residues from the reference sequence that should be kept on each side of the region spanned by sequencing reads by entering the number in the **Extra residues** field.

- **Do not include reference sequence in contig(s).** This will produce a contig data-object without the reference sequence. The contig is created in the same way as when you make an ordinary assembly (see section 18.3), but the reference sequence is omitted in the resulting contig. In the assembly process the reference sequence is only used as a scaffold for alignment. This option is useful when performing assembly with a reference sequence that is not closely related to the sequencing reads.
  - **Conflicts resolved with.** If there is a conflict, i.e. a position where there is disagreement about the residue (A, C, T or G), you can specify how the contig sequence should reflect this conflict:
    - **Unknown nucleotide (N).** The contig will be assigned an 'N' character in all positions with conflicts.
    - **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the reads. For an overview of ambiguity codes, see Appendix I.
    - **Vote (A, C, G, T).** The conflict will be solved by counting instances of each nucleotide and then letting the majority decide the nucleotide in the contig. In case of equality, ACGT are given priority over one another in the stated order.

    Note, that conflicts will always be highlighted no matter which of the options you choose. Furthermore, each conflict will be marked as annotation on the contig sequence and will be present if the contig sequence is extracted for further analysis. As a result, the details of any experimental heterogeneity can be maintained and used when the result of single-sequence analyzes is interpreted.

When the parameters have been adjusted, click **Next**, to see the dialog shown in figure 18.8


Assemble Sequences to Reference
1. Select some nucleotide sequences
2. Set reference parameters
3. Set algorithm parameters
Set algorithm parameters
?
Previous
Next
Finish
Cancel

Alignment options

Minimum aligned read length: 50

Alignment stringency: Medium

Trimming options

□ Use existing trim information

(Generally not necessary since a reference sequence is used)

Output options

□ Show tabular view of contigs

Figure 18.8: *Different options for the output of the assembly.*

In this dialog, you can specify more options:

- **Minimum aligned read length.** The minimum number of nucleotides in a read which must be successfully aligned to the contig. If this criteria is not met by a read, this is excluded from the assembly.
- **Alignment stringency.** Specifies the stringency of the scoring function used by the alignment step in the contig assembly algorithm. A higher stringency level will tend to produce contigs

with less ambiguities but will also tend to omit more sequencing reads and to generate more and shorter contigs. Three stringency levels can be set:

  - **Low.**
  - **Medium.**
  - **High.**

- **Use existing trim information.** When using a reference sequence, trimming is generally not necessary, but if you wish to use trimming you can check this box. It requires that the sequence reads have been trimmed beforehand (see section 18.2 for more information about trimming).
- **Show tabular view of contigs.** A contig can be shown both in a graphical as well as a tabular view. If you select this option, a tabular view of the contig will also be opened (Even if you do not select this option, you can show the tabular view of the contig later on by clicking **Show** () and selecting **Table** ().) For more information about the tabular view of contigs, see section 18.6.7.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will start the assembly process. See section 18.6 on how to use the resulting contigs.

18.5 Add sequences to an existing contig

This section describes how to assemble sequences to an existing contig. This feature can be used for example to provide a steady work-flow when a number of exons from the same gene are sequenced one at a time and assembled to a reference sequence.

Note that the new sequences will be added to the existing contig which will not be extended. If the new sequences extend beyond the existing contig, they will be cut off.

To start the assembly:

select one contig and a number of sequences | **Toolbox in the Menu Bar** | **Sequencing Data Analyses** () | **Add Sequences to Contig** ()

or **right-click in the empty white area of the contig** | **Add Sequences to Contig** ()

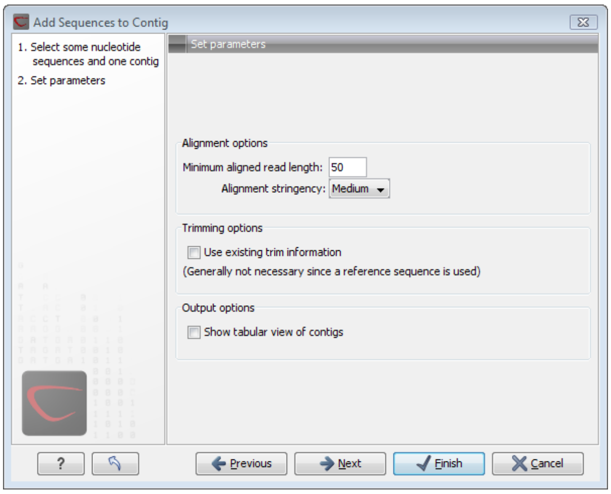
This opens a dialog where you can alter your choice of sequences which you want to assemble. You can also add sequence lists.

When the elements are selected, click **Next**, and you will see the dialog shown in figure 18.9

The options in this dialog are similar to the options that are available when assembling to a reference sequence (see section 18.4).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will start the assembly process. See section 18.6 on how to use the resulting contig.

Note that the new sequences will be added to the existing contig which will not be extended. If the new sequences extend beyond the existing contig, they will be cut off.



Add Sequences to Contig

1. Select some nucleotide sequences and one contig
2. Set parameters

Set parameters

Alignment options

Minimum aligned read length: 50

Alignment stringency: Medium

Trimming options

□ Use existing trim information

(Generally not necessary since a reference sequence is used)

Output options

□ Show tabular view of contigs

Figure 18.9: *Setting assembly parameters when assembling to an existing contig.*

18.6 View and edit read mappings

The result of the mapping process is one or more read mappings where the sequence reads have been aligned (see figure 18.10). If multiple reference sequences were used, this information will be in a table where the actual visual mapping can be opened by double-clicking.

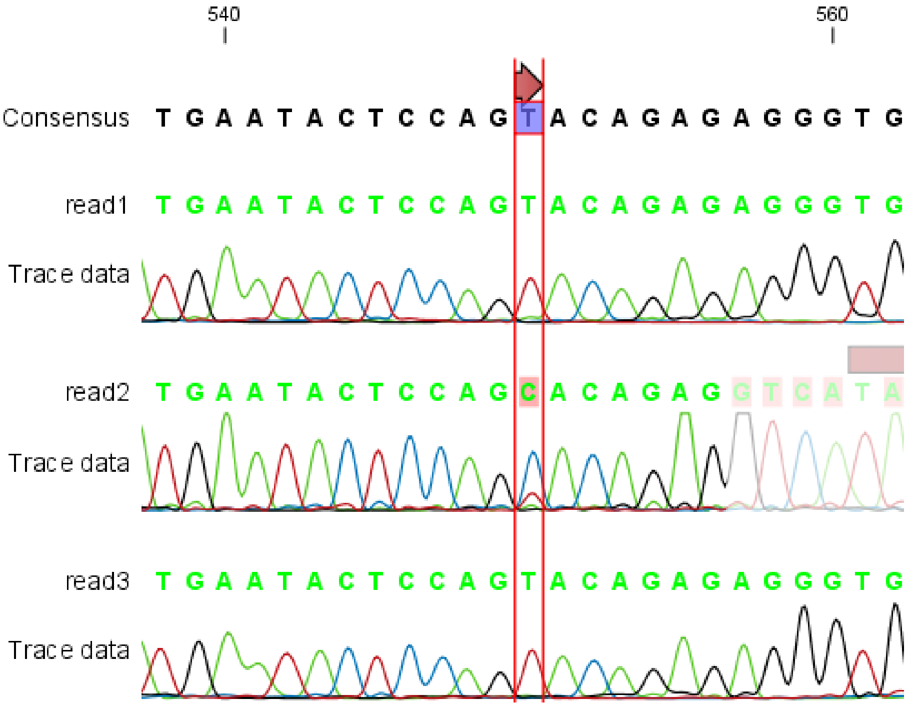

540
560
Consensus
T G A A T A C T C C A G T A C A G A G A G G G T G
read1
T G A A T A C T C C A G T A C A G A G A G G G T G
Trace data
read2
T G A A T A C T C C A G C A C A G A G G T C A T A
Trace data
read3
T G A A T A C T C C A G T A C A G A G A G G G T G
Trace data

Figure 18.10: *The view of a read mapping. Notice that you can zoom to a very detailed level in read mappings.*

You can see that color of the residues and trace at the end of one of the reads has been faded. This indicates, that this region has not contributed to the mapping. This may be due to trimming before or during the assembly or due to misalignment to the other reads.

You can easily adjust the trimmed area to include more of the read in the mapping: simply drag the edge of the faded area as shown in figure 18.11.

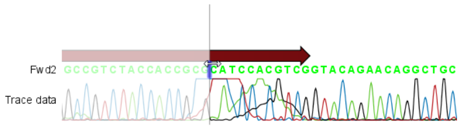


Figure 18.11: *Dragging the edge of the faded area.*

Note! This is only possible when you can see the residues on the reads. This means that you need to have zoomed in to 100% or more and chosen **Compactness** levels "Not compact" or "Low". Otherwise the handles for dragging are not available (this is done in order to make the visual overview more simple).

If reads have been reversed, this is indicated by red. Otherwise, the residues are colored green. The colors can be changed in the **Side Panel** as described in section 18.6.1

If you find out that the reversed reads should have been the forward reads and vice versa, you can reverse complement the whole mapping(imagine flipping the whole mapping):

right-click in the empty white area of the mapping | **Reverse Complement**

18.6.1 View settings in the Side Panel

Apart from this the view resembles that of alignments (see section 22.2) but has some extra preferences in the **Side Panel**: [1]

- **Read layout.** A new preference group located at the top of the **Side Panel**:
  - **Gather sequences at top.** Enabling this option affects the view that is shown when scrolling horizontally. If selected, the sequence reads which did not contribute to the visible part of the mapping will be omitted whereas the contributing sequence reads will automatically be placed right below the reference.
  - **Show sequence ends.** Regions that have been trimmed are shown with faded traces and residues. This illustrates that these regions have been ignored during the assembly.
  - **Find Conflict.** Clicking this button selects the next position where there is an conflict between the sequence reads. Residues that are different from the reference are colored (as default), providing an overview of the conflicts. Since the next conflict is automatically selected it is easy to make changes. You can also use the Space key to find the next conflict.
- **Sequence layout.** There is one additional parameter regarding the sequence layout:
  - **Compactness**: In the **Sequence Layout** view preferences, you can control the level of sequence detail to be displayed:
    - **Not compact.** The normal setting with full detail.
    - **Low.** Hides the trace data and puts the reads' annotations on the sequence.
    - **Medium.** The labels of the reads and their annotations are hidden, and the residues of the reads can not be seen.
    - **Compact.** Even less space between the reads.

[1] Note that for interpretation of mappings with large amounts of data, have a look at section 19.9

  - Furthermore, it is not possible to wrap mappings as you can do with alignments.

- **Alignment info.** There is one additional parameter:
  - **Coverage**: Shows how many sequence reads that are contributing information to a given position in the mapping. The level of coverage is relative to the overall number of sequence reads.
    - **Foreground color.** Colors the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage.
    - **Background color.** Colors the background of the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage
    - **Graph.** The coverage is displayed as a graph (Learn how to export the data behind the graph in section 7.4).
      - **Height.** Specifies the height of the graph.
      - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
      - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. If a Color bar is chosen, the color box is replaced by a gradient color box as described under Foreground color.

- **Residue coloring.** There is one additional parameter:
  - **Sequence colors.** This option lets you use different colors for the reads.
    - **Main**. The color of the consensus and reference sequence. Black per default.
    - **Forward**. The color of forward reads (single reads). Green per default.
    - **Reverse**. The color of reverse reads (single reads). Red per default.
    - **Paired**. The color of paired reads. Blue per default. Note that reads from **broken pairs** are colored according to their Forward/Reverse orientation or as a Non-specific match, but with a darker nuance than ordinary single reads.
    - **Non-specific matches**. When a read would have matched equally well another place in the mapping, it is considered a non-specific match. This color will "overrule" the other colors. Note that if you are mapping with several reference sequences, a read is considered a double match when it matches more than once *across all the contigs/references*. A non-specific match is yellow per default.

Beside from these preferences, all the functionalities of the alignment view are available. This means that you can e.g. add annotations (such as SNP annotations) to regions of interest.

However, some of the parameters from alignment views are set at a different default value in the view of contigs. Trace data of the sequencing reads are shown if present (can be enabled and disabled under the Nucleotide info preference group), and the **Color different residues** option is also enabled in order to provide a better overview of conflicts (can be changed in the Alignment info preference group).

18.6.2 Editing the read mapping

When editing mappings, you are typically interested in confirming or changing single bases, and this can be done simply by:

selecting the base | typing the right base

Some users prefer to use lower-case letters in order to be able to see which bases were altered when they use the results later on. In *CLC Genomics Workbench* all changes are recorded in the history log (see section 8) allowing the user to quickly reconstruct the actions performed in the editing session.

There are three shortcut keys for easily finding the positions where there are conflicts:

- Space bar: Finds the *next* conflict.
- "." (punctuation mark key): Finds the *next* conflict.
- "," (comma key): Finds the *previous* conflict.

In the mapping view, you can use **Zoom in** () to zoom to a greater level of detail than in other views (see figure 18.10). This is useful for discerning the trace curves.

If you want to replace a residue with a gap, use the **Delete** key.

If you wish to edit a selection of more than one residue:

right-click the selection | Edit Selection ()

This will show a warning dialog, but you can choose never to see this dialog again by clicking the checkbox at the bottom of the dialog.

Note that for mappings with more than 1000 reads, you can only do single-residue replacements (you can't delete or edit a selection).

18.6.3 Sorting reads

If you wish to change the order of the sequence reads, simply drag the label of the sequence up and down. Note that this is not possible if you have chosen **Gather sequences at top** in the **Side Panel**.

You can also sort the reads by right-clicking a sequence label and choose from the following options:

- **Sort Reads by Alignment Start Position.** This will list the first read in the alignment at the top etc.
- **Sort Reads by Name.** Sort the reads alphabetically.
- **Sort Reads by Length.** The shortest reads will be listed at the top.

18.6.4 Read conflicts

When the mapping is created, conflicts between the reads are annotated on the consensus sequence. The definition of a conflict is *a position where at least one of the reads have a different residue*.

A conflict can be in two states:

- **Conflict**. Both the annotation and the corresponding row in the Table (▦) are colored **red.**
- **Resolved**. Both the annotation and the corresponding row in the Table (▦) are colored **green**.

The conflict can be resolved by correcting the deviating residues in the reads as described above.

A fast way of making all the reads reflect the consensus sequence is to select the position in the consensus, right-click the selection, and choose **Transfer Selection to All Reads**.

The opposite is also possible: make a selection on one of the reads, right click, and **Transfer Selection to Contig Sequence**.

18.6.5 Output from the mapping

Due to the integrated nature of *CLC Genomics Workbench* it is easy to use the consensus sequences as input for additional analyzes. There are two options:

> **right-click the name of the consensus sequence (to the left) | Open Copy of Sequence | Save (💾) the new sequence**
>
> **right-click the name of the consensus sequence (to the left) | Open This Sequence**

The first option will create a copy of the sequence which can be saved and used independently. The second option will not crate a new sequence but simply let you see the sequence in a sequence view. This means that the sequence still "belong" to the mapping and will be saved together with the mapping. It also means that if you add annotations to the sequence, they will be shown in the mapping view as well. This can be very convenient e.g. for Primer design (▥).

In addition to the two options described above, you can also open the consensus sequence including gaps (**Open Copy of Sequence Including Gaps**). This will replace all gaps with Ns.

If you wish to BLAST the consensus sequence, simply select the whole contig for your BLAST search. It will automatically extract the consensus sequence and perform the BLAST search.

In order to preserve the history of the changes you have made to the contig, the contig itself should be saved from the contig view, using either the save button (💾) or by dragging it to the **Navigation Area**.

18.6.6 Extract parts of a mapping

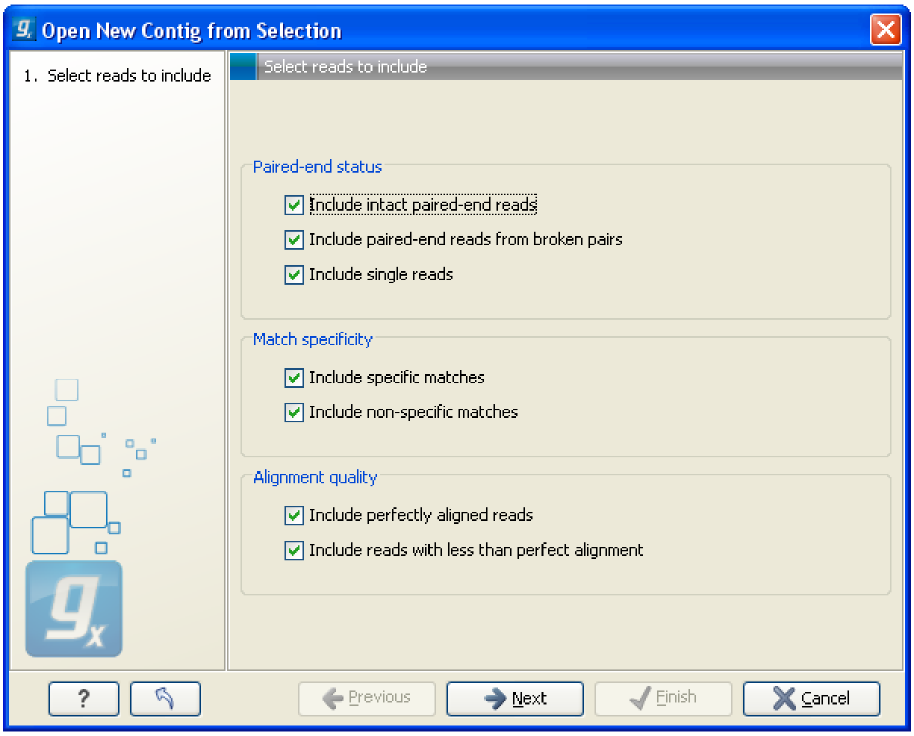
Sometimes it is useful to extract part of a mapping for in-depth analysis. This could be the case if you have performed an assembly of several genes and you want to look at a particular gene or region in isolation.

This is possible through the right-click menu of the reference or consensus sequence:

> **Select on the reference or consensus sequence the part of the contig to extract | Right-click | Extract from Selection**

This will present the dialog shown in figure 18.12.

The purpose of this dialog is to let you specify what kind of reads you want to include. Per default all reads are included. The options are:



Paired-end status

☑ Include intact paired-end reads

☑ Include paired-end reads from broken pairs

☑ Include single reads

Match specificity

☑ Include specific matches

☑ Include non-specific matches

Alignment quality

☑ Include perfectly aligned reads

☑ Include reads with less than perfect alignment

Figure 18.12: *Selecting the reads to include.*

Paired status **Include intact paired reads** When paired reads are placed within the paired distance specified, they will fall into this category. Per default, these reads are colored in blue.

Include paired reads from broken pairs When a pair is broken, either because only one read in the pair matches, or because the distance or relative orientation is wrong, the reads are placed and colored as single reads, but you can still extract them by checking this box.

Include single reads This will include reads that are marked as single reads (as opposed to paired reads). Note that paired reads that have been broken during assembly are not included in this category. Single reads that come from trimming paired sequence lists are included in this category.

Match specificity **Include specific matches** Reads that only are mapped to one position.

Include non-specific matches Reads that have multiple equally good alignments to the reference. These reads are colored yellow per default.

Alignment quality **Include perfectly aligned reads** Reads where *the full read* is perfectly aligned to the reference sequence (or consensus sequence for de novo assemblies). Note that at the end of the contig, reads may extend beyond the contig (this is not visible unless you make a selection on the read and observe the position numbering in the status bar). Such reads are not considered perfectly aligned reads because they don't align in their entire length.

Include reads with less than perfect alignment Reads with mismatches, insertions or deletions, or with unaligned nucleotides at the ends (the faded part of a read).

Note that only reads that are completely covered by the selection will be part of the new contig.

One of the benefits of this is that you can actually use this tool to extract subset of reads from a contig. An example work flow could look like this:

1. Select the whole reference sequence
2. Right-click and **Extract from Selection**
3. Choose to include only paired matches
4. Extract the reads from the new file (see section 10.7.3)

You will now have all paired reads from the original mapping in a list.

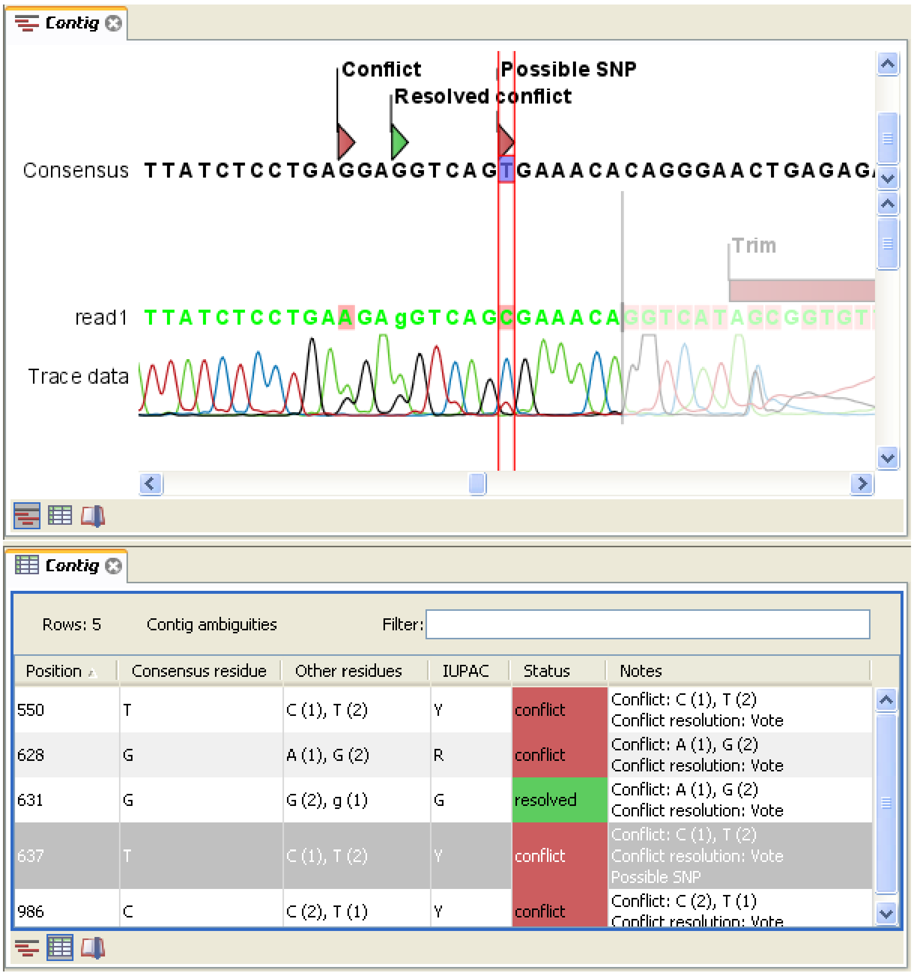
18.6.7 Variance table

In addition to the standard graphical display of a mapping as described above, you can also see a tabular overview of the conflicts between the reads by clicking the **Table ( )** icon at the bottom of the view.

This will display a new view of the conflicts as shown in figure 18.13.

The table has the following columns:

- **Position.** The position of the conflict measured from the starting point of the reference sequence.
- **Consensus residue.** The consensus's residue at this position. The residue can be edited in the graphical view, as described above.
- **Other residues.** Lists the residues of the reads. Inside the brackets, you can see the number of reads having this residue at this position. In the example in figure 18.13, you can see that at position 637 there is a 'C' in the top read in the graphical view. The other two reads have a 'T'. Therefore, the table displays the following text: 'C (1), T (2)'.
- **IUPAC.** The ambiguity code for this position. The ambiguity code reflects the residues in the reads - not in the consensus sequence. (The IUPAC codes can be found in section I.)
- **Status.** The status can either be conflict or resolved:
  - **Conflict.** Initially, all the rows in the table have this status. This means that there is one or more differences between the sequences at this position.
  - **Resolved.** If you edit the sequences, e.g. if there was an error in one of the sequences, and they now all have the same residue at this position, the status is set to *Resolved*.
- **Note.** Can be used for your own comments on this conflict. Right-click in this cell of the table to add or edit the comments. The comments in the table are associated with the conflict annotation in the graphical view. Therefore, the comments you enter in the table will also be attached to the annotation on the consensus sequence (the comments can be displayed by placing the mouse cursor on the annotation for one second - see figure 18.13). The comments are saved when you **Save** ( ).


Contig
Conflict
Possible SNP
Resolved conflict
Consensus TTATCTCCTGAGGAGGTCAGTGAAACACAGGGAACTGAGAG
Trim
read1 TTATCTCCTGAAGAgGTCAGCGAAACAGGTCATAGCGGTGT
Trace data
Contig

Rows: 5 Contig ambiguities Filter:

Position	Consensus residue	Other residues	IUPAC	Status	Notes
550	T	C (1), T (2)	Y	conflict	Conflict: C (1), T (2) Conflict resolution: Vote
628	G	A (1), G (2)	R	conflict	Conflict: A (1), G (2) Conflict resolution: Vote
631	G	G (2), g (1)	G	resolved	Conflict: A (1), G (2) Conflict resolution: Vote
637	T	C (1), T (2)	Y	conflict	Conflict: C (1), T (2) Conflict resolution: Vote Possible SNP
986	C	C (2), T (1)	Y	conflict	Conflict: C (2), T (1) Conflict resolution: Vote

Figure 18.13: *The graphical view is displayed at the top. At the bottom the conflicts are shown in a table. At the conflict at position 637, the user has entered a comment in the table. This comment is now also reflected on the tooltip of the conflict annotation in the graphical view above.*

By clicking a row in the table, the corresponding position is highlighted in the graphical view. Clicking the rows of the table is another way of navigating the mapping, apart from using the **Find Conflict** button or using the **Space bar**. You can use the up and down arrow keys to navigate the rows of the table.

18.7 Reassemble contig

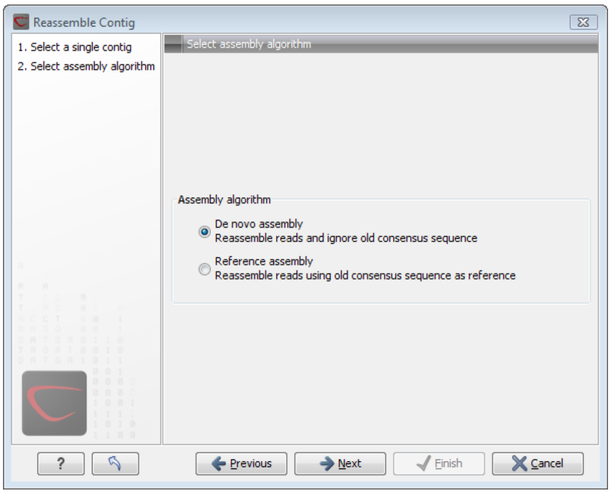
If you have edited a contig, changed trimmed regions, or added or removed reads, you may wish to reassemble the contig. This can be done in two ways:

Toolbox in the Menu Bar | **Sequencing Data Analyses** ( ) | **Reassemble Contig** ( ) | **select the contig and click Next**

or **right-click in the empty white area of the contig** | **Reassemble contig** ( )

This opens a dialog as shown in figure 18.14

In this dialog, you can choose:


Reassemble Contig
1. Select a single contig
2. Select assembly algorithm
Select assembly algorithm
?
Previous
Next
Finish
Cancel

Assembly algorithm

● De novo assembly
Reassemble reads and ignore old consensus sequence

○ Reference assembly
Reassemble reads using old consensus sequence as reference

Figure 18.14: *Re-assembling a contig.*

- **De novo assembly**. This will perform a normal assembly in the same way as if you had selected the reads as individual sequences. When you click **Next**, you will follow the same steps as described in section 18.3. The consensus sequence of the contig will be ignored.

- **Reference assembly**. This will use the consensus sequence of the contig as reference. When you click **Next**, you will follow the same steps as described in section 18.4.

When you click **Finish**, a new contig is created, so you do not lose the information in the old contig.

18.8 Secondary peak calling

CLC Genomics Workbench is able to detect secondary peaks - a peak within a peak - to help discover heterozygous mutations. Looking at the height of the peak below the top peak, the *CLC Genomics Workbench* considers all positions in a sequence, and if a peak is higher than the threshold set by the user, it will be "called".

The peak is called by changing the residue to an ambiguity character and by adding an annotation at this position.

To call secondary peaks:

> **select sequence(s)** | **Toolbox in the Menu Bar** | **Sequencing Data Analyses ()** | **Call Secondary Peaks ()**

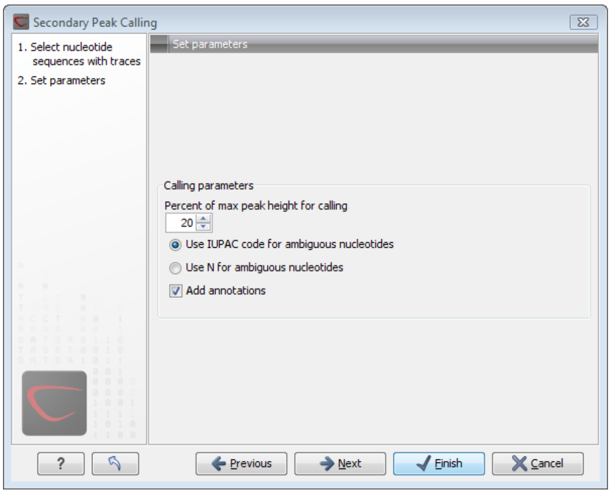
This opens a dialog where you can alter your choice of sequences.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 18.15.

The following parameters can be adjusted in the dialog:

- **Percent of max peak height for calling.** Adjust this value to specify how high the secondary peak must be to be called.



Secondary Peak Calling

1. Select nucleotide sequences with traces
2. Set parameters

Set parameters

Calling parameters

Percent of max peak height for calling: 20

● Use IUPAC code for ambiguous nucleotides

○ Use N for ambiguous nucleotides

☑ Add annotations

Figure 18.15: *Setting parameters secondary peak calling.*

- **Use IUPAC code / N for ambiguous nucleotides.** When a secondary peak is called, the residue at this position can either be replaced by an N or by a ambiguity character based on the IUPAC codes (see section I).
- **Add annotations.** In addition to changing the actual sequence, annotations can be added for each base which has been called.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will start the secondary peak calling. A detailed history entry will be added to the history specifying all the changes made to the sequence.

Chapter 19

High-throughput sequencing

Contents

The so-called Next Generation Sequencing technologies encompass a range of technologies generating huge amounts of sequencing data at a very high speed compared to traditional Sanger sequencing. The *CLC Genomics Workbench* lets you import, trim, map, assemble and analyze DNA sequence reads from all of these high-throughput sequencing machines:

- The 454 FLX System from Roche
- Illumina's Genome Analyzer
- SOLiD system from Applied Biosystems (read mapping is performed in color space, see section 19.8)
- HeliScope system from Helicos BioSciences

The *CLC Genomics Workbench* supports paired data from all platforms. Paired reads can be extremely powerful especially for sequencing repeat regions, where assembly of short reads can be difficult. Knowing that there is an approximate distance between two reads enhances the possibilities of performing a correct assembly. It also enables a wide array of new approaches to interpreting the sequencing data.

This chapter first explains how to import the sequencing data. Since these data are different from general data formats accepted by the *CLC Genomics Workbench*, it requires a special explanation. Following the import section is a description of how trimming the data based on quality scores can be performed.

The following two sections describe *de novo* assembly and read mapping, and how to interpret the results. At the end you can read about SNP detection.

19.1 Import high-throughput sequencing data

This section describes how to import data from high-throughput sequencing machines:

> **File | Import High-Throughput Sequencing Data ()**

This will bring up a list of the formats supported as shown in figure 19.1

Select the appropriate format (see descriptions below).

19.1.1 454 from Roche Applied Science

Choosing the Roche 454 import will open the dialog shown in figure 19.2.

We support import of two kinds of data from 454 GS FLX systems:

- Flowgram files (`.sff`) which contain both sequence data and quality scores amongst others. However, the flowgram information is currently not used by *CLC Genomics Workbench*. There is an extra option to make use of clipping information (this will remove parts of the sequence as specified in the .sff file).
- Fasta/qual files:

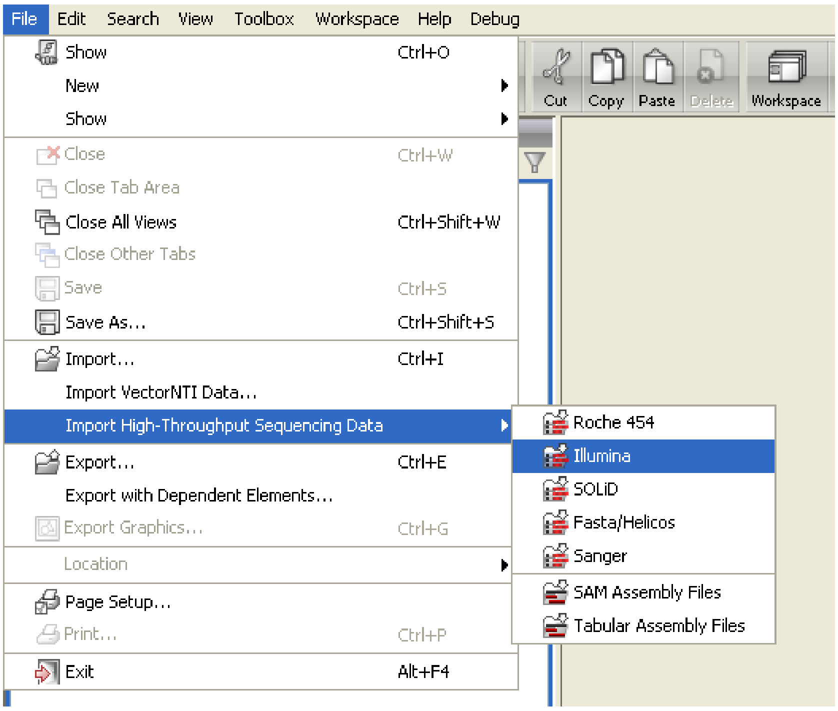
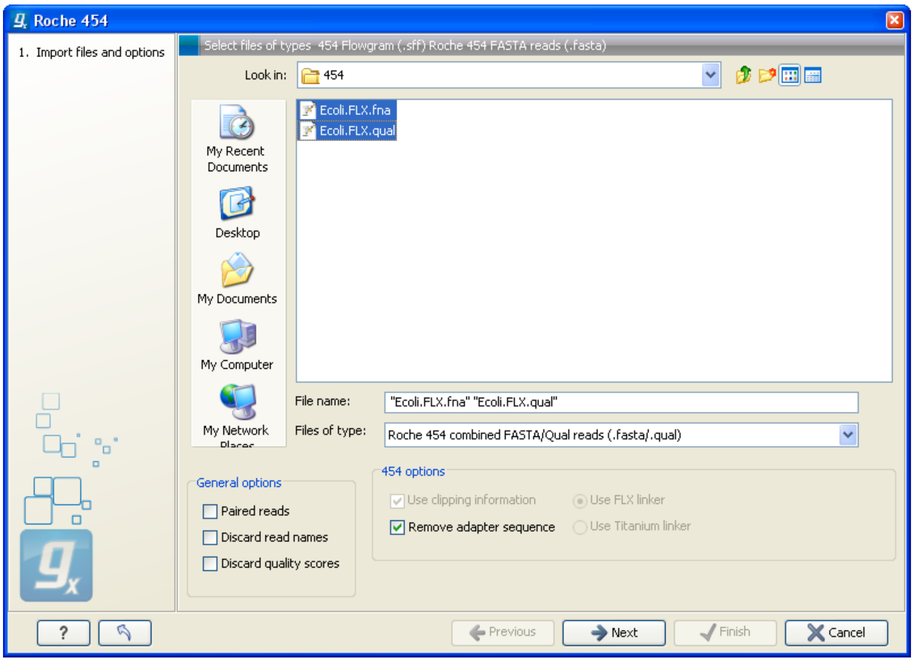


Figure 19.1: *Choosing what kind of data you wish to import.*



Look in: 454

File name: "Ecoli.FLX.fna" "Ecoli.FLX.qual"

Files of type: Roche 454 combined FASTA/Qual reads (.fasta/.qual)

General options

- □ Paired reads
- □ Discard read names
- □ Discard quality scores

454 options

- ☑ Use clipping information
- ☑ Remove adapter sequence
- ● Use FLX linker
- ○ Use Titanium linker

Figure 19.2: *Importing data from Roche 454.*

- 454 FASTA files (`.fna`) which contain the sequence data.
- Quality files (`.qual`) which contain the quality scores.

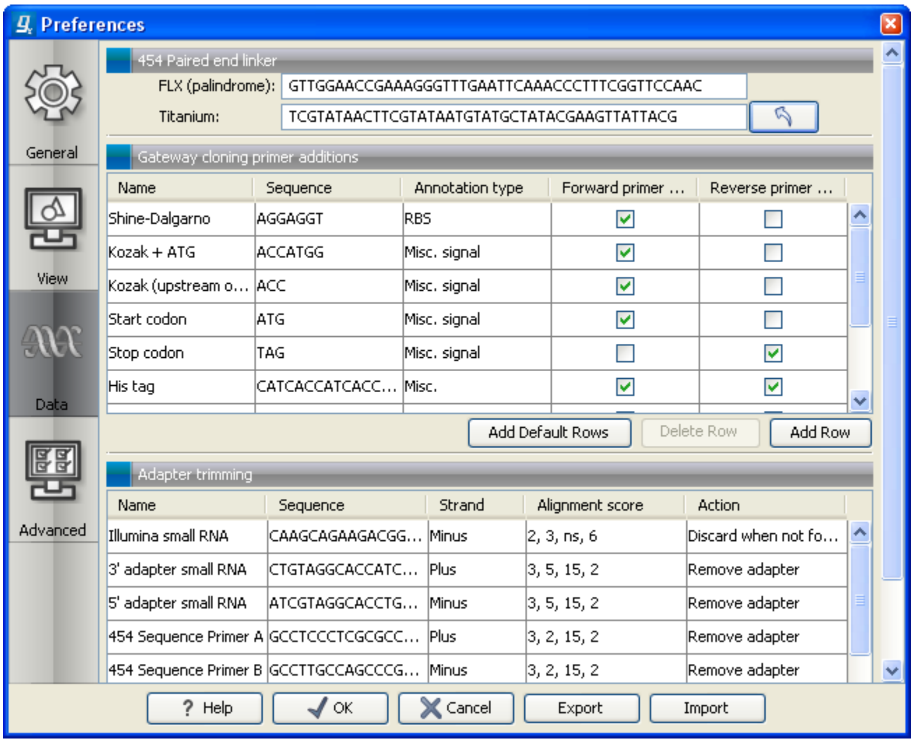
For all formats, compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. The paired protocol for 454 entails that the forward and reverse reads are separated by a linker sequence. During import of paired data, the linker sequence is removed and the forward and reverse reads are separated and put into the same sequence list (their status as forward and reverse reads is preserved). You can change the linker sequence in the **Preferences** (in the **Edit** menu) under **Data**. Since the linker for the FLX and Titanium versions are different, you can choose the appropriate protocol during import, and in the preferences you can supply a linker for both platforms (see figure 19.3. Note that since the FLX linker is palindromic, it will only be searched on the plus strand, whereas the Titanium linker will be found on both strands. Some of the sequences may not have the linker in the middle of the sequence, and in that case the partial linker sequence is still removed, and the single read is put into a separate sequence list. Thus when you import 454 paired data, you may end up with two sequence lists: one for paired reads and one for single reads. Note that for de novo assembly projects, only the paired list should be used since the single reads list may contain reads where there is still a linker sequence present but only partially due to sequencing errors.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. If you have selected the fna/qual option and choose to discard quality scores, you do not need to select a .qual file.



454 Paired end linker

FLX (palindrome): GTTGGAACCGAAAGGGTTTGAATTCAAACCCTTTCGGTTCCAAC

Titanium: TCGTATAACTTCGTATAATGTATGCTATACGAAGTTATTACG

Gateway cloning primer additions

Name	Sequence	Annotation type	Forward primer ...	Reverse primer ...
Shine-Dalgarno	AGGAGGT	RBS	☑	□
Kozak + ATG	ACCATGG	Misc. signal	☑	□
Kozak (upstream o...	ACC	Misc. signal	☑	□
Start codon	ATG	Misc. signal	☑	□
Stop codon	TAG	Misc. signal	□	☑
His tag	CATCACCATCACC...	Misc.	☑	☑

Adapter trimming

Name	Sequence	Strand	Alignment score	Action
Illumina small RNA	CAAGCAGAAGACGG...	Minus	2, 3, ns, 6	Discard when not fo...
3' adapter small RNA	CTGTAGGCACCATC...	Plus	3, 5, 15, 2	Remove adapter
5' adapter small RNA	ATCGTAGGCACCTG...	Minus	3, 5, 15, 2	Remove adapter
454 Sequence Primer A	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Remove adapter
454 Sequence Primer B	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Remove adapter

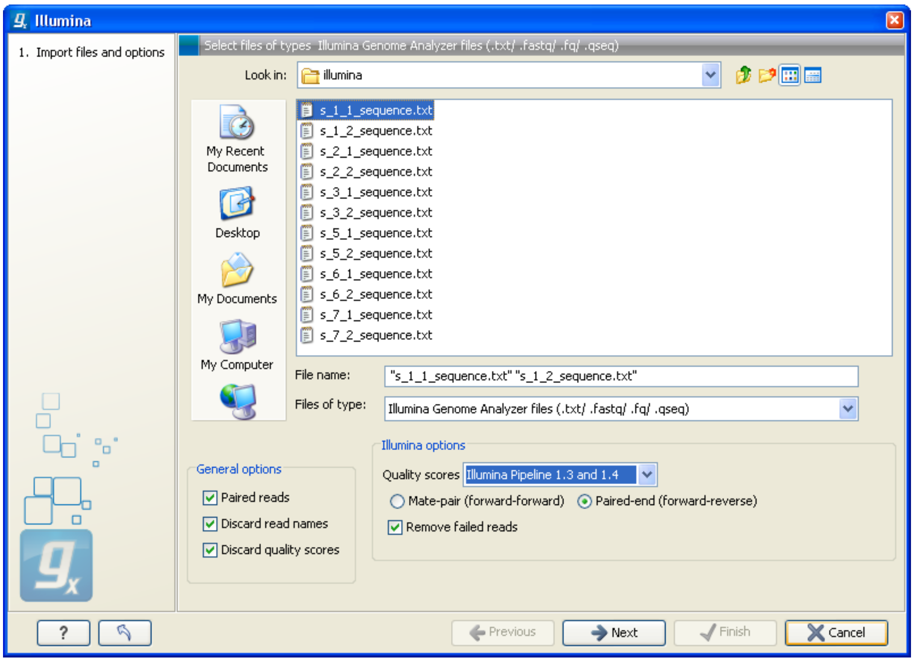
Figure 19.3: *Specifying linkers for 454 import.*

Note! During import, partial adapter sequences are removed (TCAG and ATGC), and if the full sequencing adapters GCCTTGCCAGCCCGCTCAG, GCCTCCCTCGCGCCATCAG or their reverse complements are found, they are also removed (including tailing Ns). If you do not wish to remove the adapter sequences (e.g. if they have already been removed by other software), please uncheck the **Remove adapter sequence** option.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 9.1).

19.1.2 Illumina Genome Analyzer from Illumina

Choosing the Illumina import will open the dialog shown in figure 19.4.



Select files of types Illumina Genome Analyzer files (.txt/ .fastq/ .fq/ .qseq)

Look in: illumina

s_1_1_sequence.txt
s_1_2_sequence.txt
s_2_1_sequence.txt
s_2_2_sequence.txt
s_3_1_sequence.txt
s_3_2_sequence.txt
s_5_1_sequence.txt
s_5_2_sequence.txt
s_6_1_sequence.txt
s_6_2_sequence.txt
s_7_1_sequence.txt
s_7_2_sequence.txt

File name: "s_1_1_sequence.txt" "s_1_2_sequence.txt"

Files of type: Illumina Genome Analyzer files (.txt/ .fastq/ .fq/ .qseq)

General options
☑ Paired reads
☑ Discard read names
☑ Discard quality scores

Illumina options
Quality scores Illumina Pipeline 1.3 and 1.4
○ Mate-pair (forward-forward) ● Paired-end (forward-reverse)
☑ Remove failed reads

Figure 19.4: *Importing data from Illumina's Genome Analyzer.*

The file formats accepted are:

- Fastq
- Scarf
- Qseq

Note that qseq files specifies whether a read has passed a quality filter or not. If you check **Remove failed reads** these reads will be ignored during import. This option does not have any effect on the fastq and scarf formats. Paired data is supported for all formats.

For all formats, compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. For paired import, you can select whether the data is **Paired-end** or **Mate-pair**. For paired-end data, the Workbench expects the forward reads to be in one file and the reverse reads in another. The file with forward reads should end with `_1_sequence.txt` or `_1_qseq.txt` and the file with reverse reads should end with `_2_sequence.txt` or `_2_qseq.txt` . Typically, the files are named as shown in figure 19.4 where the forward file is called `s_1_1_sequence.txt` and the reverse is called `s_1_2_sequence.txt`. Within each file, forward reads have a `/1` at the end of the read name, and reverse reads have a `/2`. For mate-pair data, the file naming and structure is the same, except that the orientation of the reads is expected to be forward-forward rather than forward-reverse.

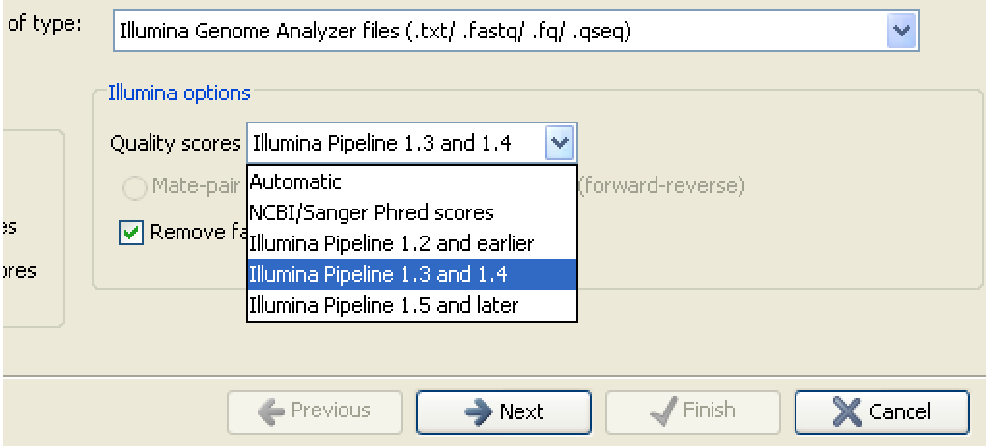
- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard quality scores to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. Read more about the quality scores of Illumina below.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 9.1).

Quality scores in the Illumina platform

The quality scores in the FASTQ format come in different versions. You can read more about the FASTQ format at http://en.wikipedia.org/wiki/FASTQ_format. When you select to import Illumina data and click **Next** there is an option to use different quality score schemes at the bottom of the dialog (see figure 19.5).



of type: Illumina Genome Analyzer files (.txt/ .fastq/ .fq/ .qseq)

Illumina options

Quality scores: Illumina Pipeline 1.3 and 1.4

Figure 19.5: *Selecting the quality score scheme.*

There are three options:

- **Automatic**. Choosing this option, the Workbench attempts to automatically detect the quality score format. Sometimes this is not possible, and you have to specify the format yourself. In the cases where the Workbench is unable to determine the format, it is usually one of the Illumina Pipeline format files. If there are characters ; < = > or ? in the quality score information, it is the old Illumina pipeline format (ASCII values 59 to 63).
- **NCBI/Sanger Phred scores**. Using a Phred scale encoded using ASCII 33 to 93.
- **Illumina Pipeline 1.2 and earlier**. Using a Solexa/Illumina scale (-5 to 40) using ASCII 59 to 104. The Workbench automatically converts these quality scores to the Phred scale on import in order to ensure a common scale for analyses across data sets from different platforms (see details on the conversion next to the sample below).
- **Illumina Pipeline 1.3 and 1.4**. Using a Phred scale using ASCII 64 to 104.
- **Illumina Pipeline 1.5 and later**. Using a Phred scale using ASCII 64 to 104. Values 0 (@) and 1 (A) are not used anymore. Value 2 (B) has special meaning and is used as a trim clipping. This means that when selecting Illumina Pipeline 1.5 and later, the *reads are automatically trimmed* when a B is encountered in the input file.

Small sample of all three kinds of files are shown below. The names of the reads have no influence on the quality score format:

NCBI/Sanger Phred scores:

```
@SRR001926.1 FC00002:7:1:111:750 length=36
TTTTTGTAAGGAGGGGGGTCATCAAAATTTGCAAAA
+SRR001926.1 FC00002:7:1:111:750 length=36
IIIIIIIIIIIIIIIIIIIIIIIIIFIIII’IB<IH
@SRR001926.7 FC00002:7:1:110:453 length=36
TTATATGGAGGCTTTAAGAGTCATAGGTTGTTCCCC
+SRR001926.7 FC00002:7:1:110:453 length=36
IIIIIIIIIII:’III?=IIIIII+&III/3I8F/&
```

Illumina Pipeline 1.2 and earlier (note the question mark at the end of line 4 - this is one of the values that are unique to the old Illumina pipeline format):

```
@SLXA-EAS1_89:1:1:672:654/1
GCTACGGAATAAAACCAGGAACAACAGACCCAGCA
+SLXA-EAS1_89:1:1:672:654/1
cccccccccccccccccccc]c``cVcZccbSYb?
@SLXA-EAS1_89:1:1:657:649/1
GCAGAAAATGGGAGTGAAAATCTCCGATGAGCAGC
+SLXA-EAS1_89:1:1:657:649/1
ccccccccccbccbccb``cccbcccZcc`^bR^`
```

The formulas used for converting the special Solexa-scale quality scores to Phred-scale:

$Q_{phred} = -10 \log_{10} p$

$Q_{solexa} = -10 \log_{10} \frac{p}{1-p}$

A sample of the quality scores of the Illumina Pipeline 1.3 and 1.4:

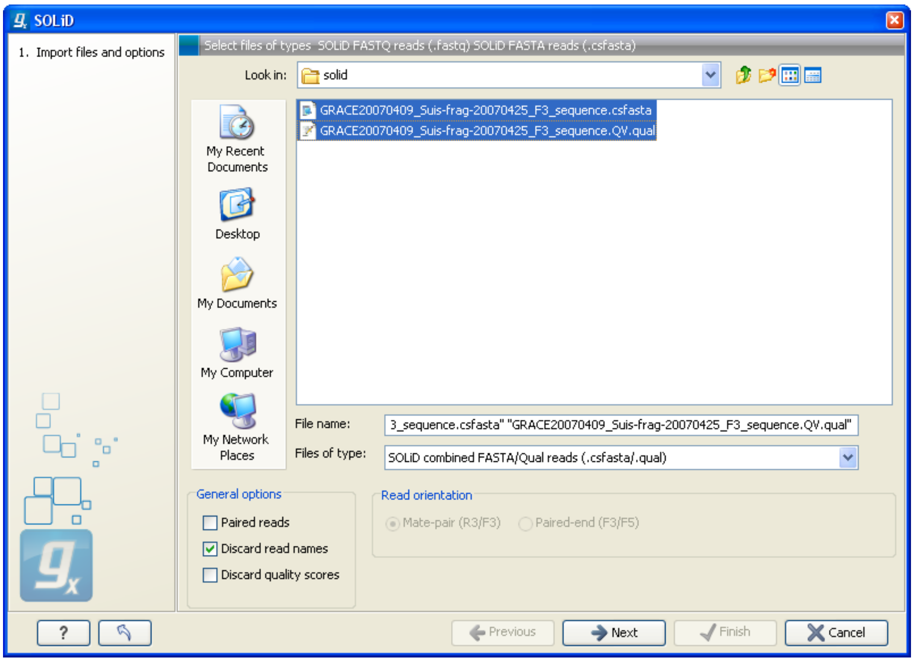
```
@HWI-E4_9_30WAF:1:1:8:178
GCCAGCGGCGCAAAATGNCGGCGGCGATGACCTTC
+HWI-E4_9_30WAF:1:1:8:178
babaaaa\abababaaaaREXabaaaaaaaaaaaaaa
@HWI-E4_9_30WAF:1:1:8:1689
GATGGAGATCTCGACCTNATAGGTGCCCTCATCGG
+HWI-E4_9_30WAF:1:1:8:1689
aab`]_aaaaaaaaaa[ER`abaaa\aaaaaaaa[
```

Note that it is not possible to see from that data itself that it is actually not Illumina Pipeline 1.2 and earlier, since they use the same range of ASCII values.

To learn more about ASCII values, please see http://en.wikipedia.org/wiki/Ascii#ASCII_printable_characters.

19.1.3 SOLiD from Applied Biosystems

Choosing the SOLiD import will open the dialog shown in figure 19.6.



SOLiD

1. Import files and options

Select files of types SOLiD FASTQ reads (.fastq) SOLiD FASTA reads (.csfasta)

Look in: solid

GRACE20070409_Suis-frag-20070425_F3_sequence.csfasta
GRACE20070409_Suis-frag-20070425_F3_sequence.QV.qual

File name: 3_sequence.csfasta" "GRACE20070409_Suis-frag-20070425_F3_sequence.QV.qual"

Files of type: SOLiD combined FASTA/Qual reads (.csfasta/.qual)

General options
- □ Paired reads
- ☑ Discard read names
- □ Discard quality scores

Read orientation
- ● Mate-pair (R3/F3)
- ○ Paired-end (F3/F5)

Figure 19.6: *Importing data from SOLiD from Applied Biosystems.*

The file format accepted is the csfasta format which is the color space version of fasta format. If you want to import quality scores, a qual files should also be provided. The reads in a csfasta file look like this:

```
>2_14_26_F3
T01121312220022112303211122102121013133222101
>2_14_192_F3
T11002122110031003012002203222211132102211222 3
>2_14_233_F3
T01100133231112121231202231020331220113211122 3
```

```
>2_14_294_F3
T2130121323000000021323212232.03300033102330332
```

All reads start with a T which specifies the right phasing of the color sequence.

If a reads has a . as you can see in the last read in the example above, it means that the color calling was ambiguous (this would have been an N if we were in base space). In this case, the Workbench simply cuts off the rest of the read, since there is now way to know the right phase of the rest of the colors in the read. If the read starts with a dot, it is not imported. If all reads start with a dot, a warning dialog will be displayed.

When the example above is imported into the Workbench, it looks as shown in figure 19.7.

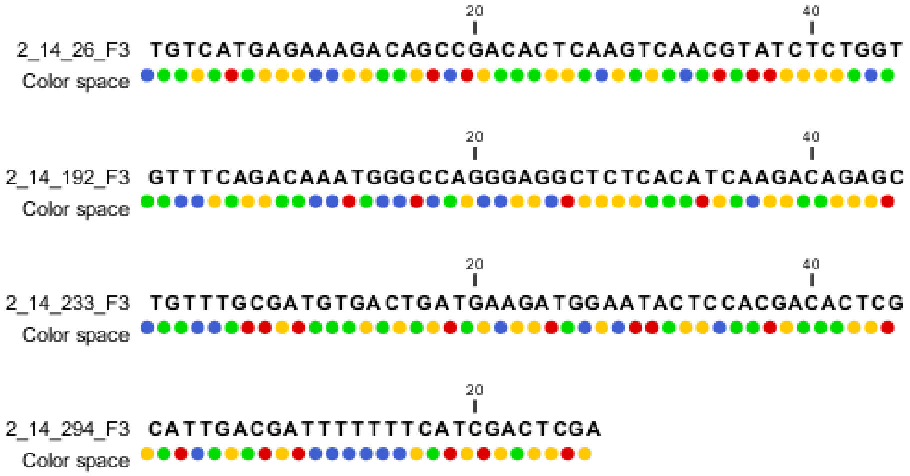

20
40
2_14_26_F3
TGTCATGAGAAAGACAGCCGACACTCAAGTCAACGTATCTCTGGT
Color space
20
40
2_14_192_F3
GTTTCAGACAAATGGGCCAGGGAGGCTCTCACATCAAGACAGAGC
Color space
20
40
2_14_233_F3
TGTTTGCGATGTGACTGATGAAGATGGAATACTCCACGACACTCG
Color space
20
2_14_294_F3
CATTGACGATTTTTTTCATCGACTCGA
Color space

Figure 19.7: *Importing data from SOLiD from Applied Biosystems. Note that the fourth read is cut off so that the color following the dot are not included*

For more information about color space, please see section 19.8.

In addition to the native csfasta format used by SOLiD, you can also input data in fastq format. This is particularly useful for data downloaded from the Sequence Read Archive at NCBI (http://www.ncbi.nlm.nih.gov/Traces/sra/). An example of a SOLiD fastq file is shown here with both quality scores and the color space encoding:

```
@SRR016056.1.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_130.1 length=50
T31000313121310211022312223311212113022121201332213
+SRR016056.1.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_130.1 length=50
!*%;2'%%050%'0'3%%5*.%%%),%%%%&%%%%%%'%%%%%'%%3+%%%
@SRR016056.2.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_223.1 length=50
T20002201120021211012010332211122133212331221302222
+SRR016056.2.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_223.1 length=50
!%%)%'))'&'%(((&%/&)%+(%%%&%%%%%%%%%%%%%%%+%%%%%%+'
```

For all formats, compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. When you import paired data, two different protocols are supported:

- **Mate-pair**. For mate-pair data, the reads should be in two files with `_F3` and `_R3` in front of the the file extension. The orientation of the reads is expected to be forward-forward.
- **Paired-end**. For paired-end data, the reads should be in two files with `_F3` and `_F5-PE` or `_F5-BC`. The orientation is expected to be forward-reverse.

An example of a complete list of the four files needed for a SOLiD mate-paired data set including quality scores:

```
dataset_F3.csfasta   dataset_F3.qual
dataset_R3.csfasta   dataset_R3.qual
```

or

```
dataset_F3.csfasta   dataset_F3_.QV.qual
dataset_R3.csfasta   dataset_R3_.QV.qual
```

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.
- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. If you choose to discard quality scores, you do not need to select a .qual file.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 9.1).

19.1.4 Fasta format

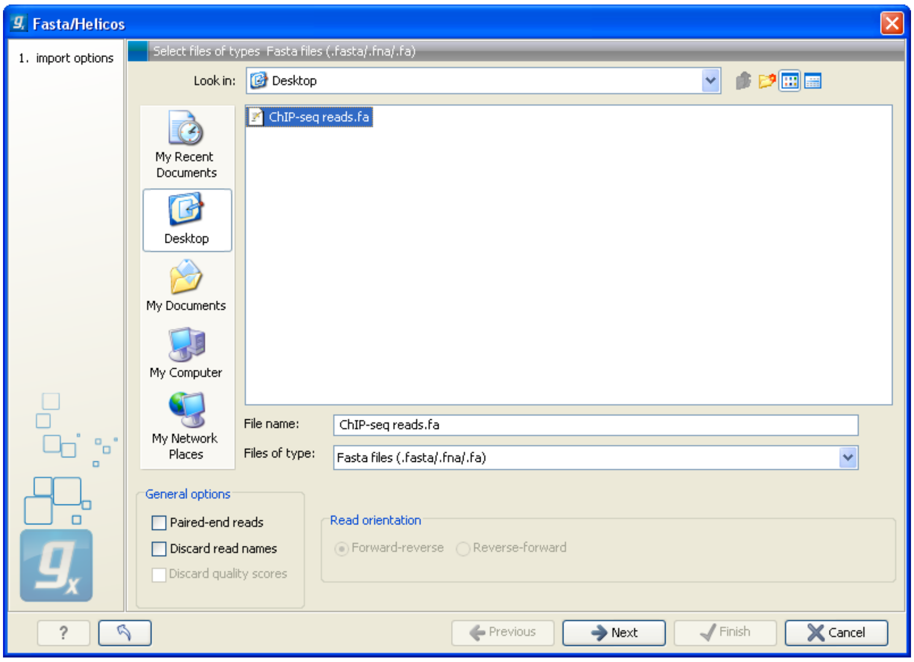
Data coming in a standard fasta format can also be imported using the standard **Import** ( ), see section 7.1. However, using the special high-throughput sequencing data import is recommended since the data is imported in a "leaner" format than using the standard import. This also means that all descriptions from the fasta files are ignored (usually there are none anyway for this kind of data).

The dialog for importing data in fasta format is shown in figure 19.8.

Compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. For paired import, the Workbench expects the forward reads to be in one file and the reverse reads in another. The Workbench will sort the files before import and then assume that the first and second file belong together, and that the third and fourth file



Select files of types Fasta files (.fasta/.fna/.fa)

Look in: Desktop

ChIP-seq reads.fa

File name: ChIP-seq reads.fa

Files of type: Fasta files (.fasta/.fna/.fa)

General options

☐ Paired-end reads

☐ Discard read names

☐ Discard quality scores

Read orientation

● Forward-reverse ○ Reverse-forward

Figure 19.8: *Importing data in fasta format.*

belong together etc. At the bottom of the dialog, you can choose whether the ordering of the files is **Forward-reverse** or **Reverse-forward**. As an example, you could have a data set with two files: `sample1_fwd` containing all the forward reads and `sample1_rev` containing all the reverse reads. In each file, the reads have to match each other, so that the first read in the `fwd` list should be paired with the first read in the `rev` list. Note that you can specify the insert sizes when running mapping and assembly. If you have data sets with different insert sizes, you should import each data set individually in order to be able to specify different insert sizes.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. This option is not relevant for fasta import, since quality scores are not supported.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 9.1).

19.1.5 Sanger sequencing data

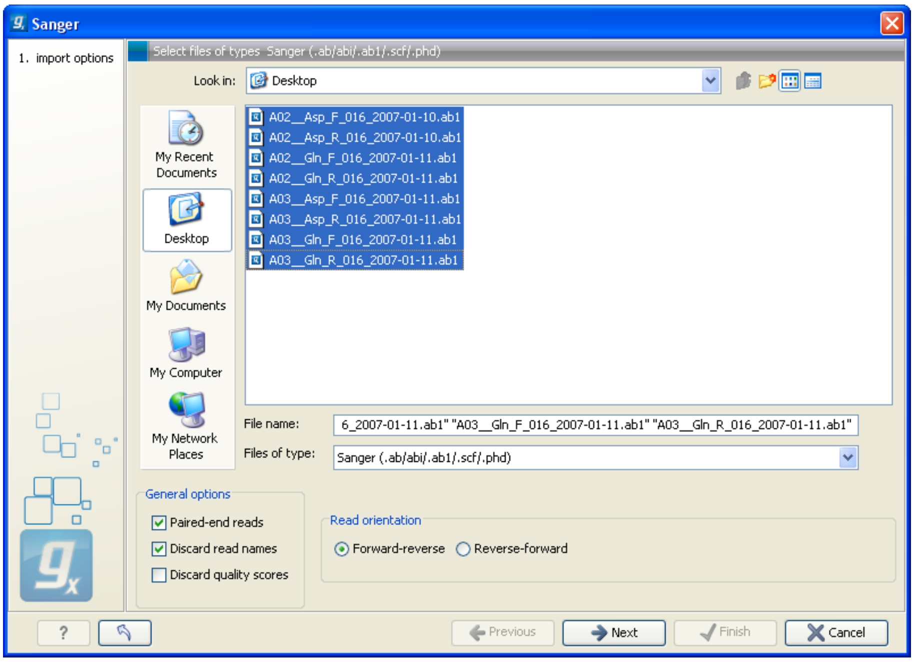
Although traditional sequencing data (with chromatogram traces like abi files) is usually imported using the standard **Import** (📂), see section 7.1, this option has also been included in the High-Throughput Sequencing Data import. It is designed to handle import of large amounts of sequences, and there are three differences from the standard import:

- All the sequences will be put in one sequence list (instead of single sequences).
- The chromatogram traces will be removed (quality scores remain). This is done to improve performance, since the trace data takes up a lot of disk space and significantly impacts speed and memory consumption for further analysis.
- Paired data is supported.

With the standard import, it is practically impossible to import up to thousands of trace files and use them in an assembly. With this special High-Throughput Sequencing import, there is no limit. The import formats supported are the same: ab, abi, ab1, scf and phd.

For all formats, compressed data in gzip format is also supported (.gz).

The dialog for importing data Sanger sequencing data is shown in figure 19.9.



Select files of types Sanger (.ab/abi/.ab1/.scf/.phd)

Look in: Desktop

A02__Asp_F_016_2007-01-10.ab1
A02__Asp_R_016_2007-01-10.ab1
A02__Gln_F_016_2007-01-11.ab1
A02__Gln_R_016_2007-01-11.ab1
A03__Asp_F_016_2007-01-11.ab1
A03__Asp_R_016_2007-01-11.ab1
A03__Gln_F_016_2007-01-11.ab1
A03__Gln_R_016_2007-01-11.ab1

File name: 6_2007-01-11.ab1" "A03__Gln_F_016_2007-01-11.ab1" "A03__Gln_R_016_2007-01-11.ab1"

Files of type: Sanger (.ab/abi/.ab1/.scf/.phd)

General options

☑ Paired-end reads
☑ Discard read names
☐ Discard quality scores

Read orientation

● Forward-reverse ○ Reverse-forward

Figure 19.9: *Importing data from Sanger sequencing.*

The **General options** to the left are:

- **Paired reads**. The Workbench will sort the files before import and then assume that the first and second file belong together, and that the third and fourth file belong together etc. At the bottom of the dialog, you can choose whether the ordering of the files is **Forward-reverse** or **Reverse-forward**. As an example, you could have a data set with two files: `sample1_fwd` for the the forward read and `sample1_rev` for the reverse reads. Note that you can specify the insert sizes when running the mapping and the assembly. If you have data sets with different insert sizes, you should import each data set individually in order to be able to specify different insert sizes.
- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 9.1).

19.1.6 General notes on handling paired data

During import, paired data is processed by the *CLC Genomics Workbench* and standardized so that after import the reads are always forward-reverse. So the first read is forward and the second reverse. This means that all subsequent analyses do not need to take differences in orientation into account. Both forward and reverse reads will be stored in the same sequence list. The forward and reverse reads simply alternates so that the first read is forward, the second read is the mate reverse read; the third is again forward and the fourth read is the mate reverse read. When deleting or manipulating sequence lists with paired data, be careful not break this order.

The distance between the reads in a pair has to be set during assembly (see sections 19.4 and 19.5 for more information).

The sequence list will mention "*paired*" in its name if it has been imported as a paired data set. You will also be able to see this information in the **History** ( ) of the sequence list.

19.1.7 SAM and BAM mapping files

The *CLC Genomics Workbench* supports import and export of files in SAM (Sequence Alignment/Map) and BAM format which are generic formats for storing large nucleotide sequence alignments. Read more and see the format specification at http://samtools.sourceforge.net/.

The idea behind the importer is that you import the sam/bam file which includes all the reads and then you specify one or more reference sequences which have already been imported into the Workbench. The Workbench will then combine the two to create a mapping result ( ) or mapping tables ( ). To import a SAM or BAM file:

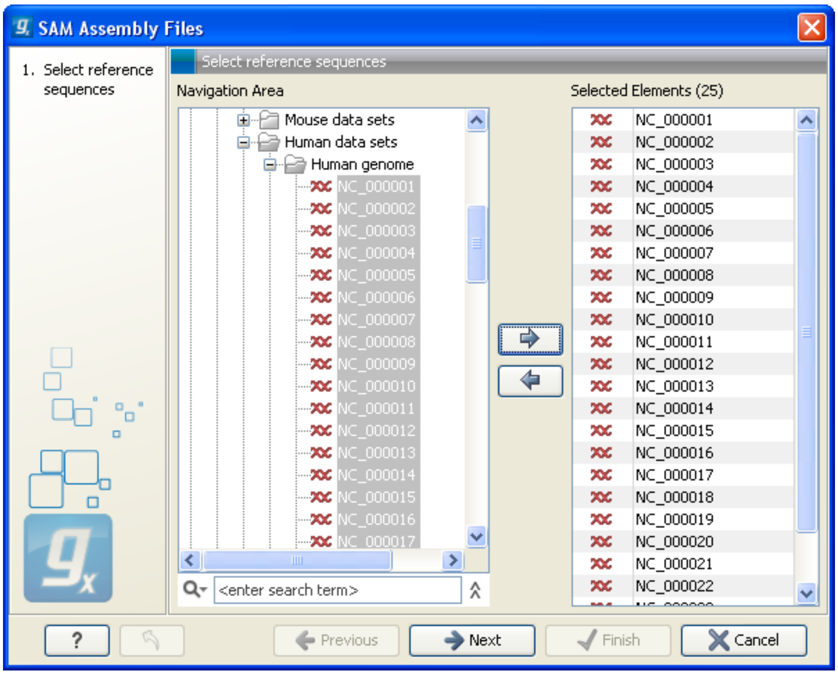
File | Import High-Throughput Sequencing Data ( ) | SAM/BAM Mapping Files ( )

This will open a dialog where you choose the reference sequences to be used as shown in figure 19.10.

Select one or more reference sequence. Note that the name of your reference sequence has to match the reference name specified in the SAM/BAM file. Click **Next**.

In this dialog, select ( ) one or more SAM/BAM files as shown in figure 19.11.

In the panel below, all the reference sequences found in the SAM/BAM file will be listed included their lengths. In addition, it is indicated in the **Status** column whether they match the reference



Navigation Area

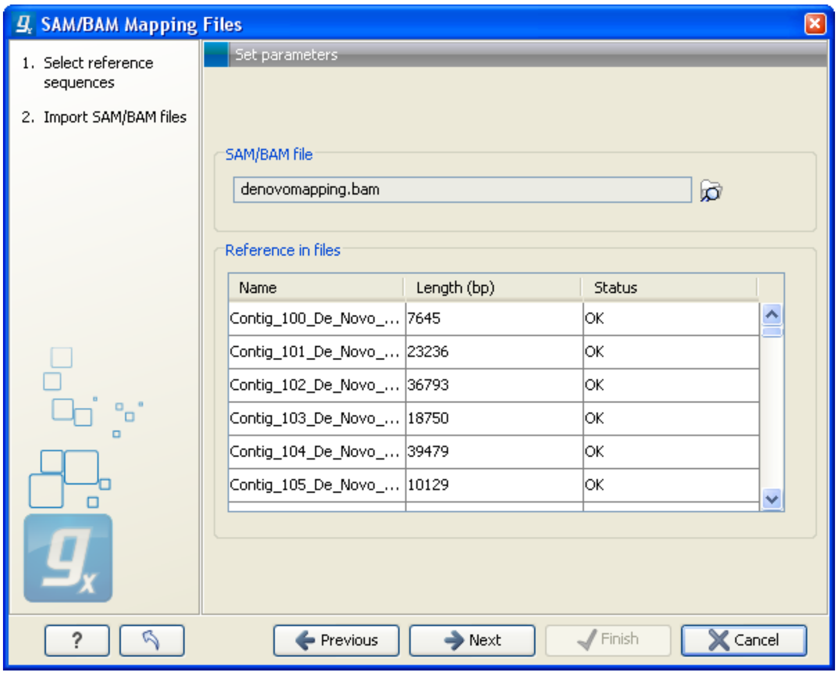
- Mouse data sets
- Human data sets
  - Human genome
    - NC_000001
    - NC_000002
    - NC_000003
    - NC_000004
    - NC_000005
    - NC_000006
    - NC_000007
    - NC_000008
    - NC_000009
    - NC_000010
    - NC_000011
    - NC_000012
    - NC_000013
    - NC_000014
    - NC_000015
    - NC_000016
    - NC_000017

<enter search term>

Selected Elements (25)

NC_000001
NC_000002
NC_000003
NC_000004
NC_000005
NC_000006
NC_000007
NC_000008
NC_000009
NC_000010
NC_000011
NC_000012
NC_000013
NC_000014
NC_000015
NC_000016
NC_000017
NC_000018
NC_000019
NC_000020
NC_000021
NC_000022

Figure 19.10: *Defining reference sequences.*



SAM/BAM file

denovomapping.bam

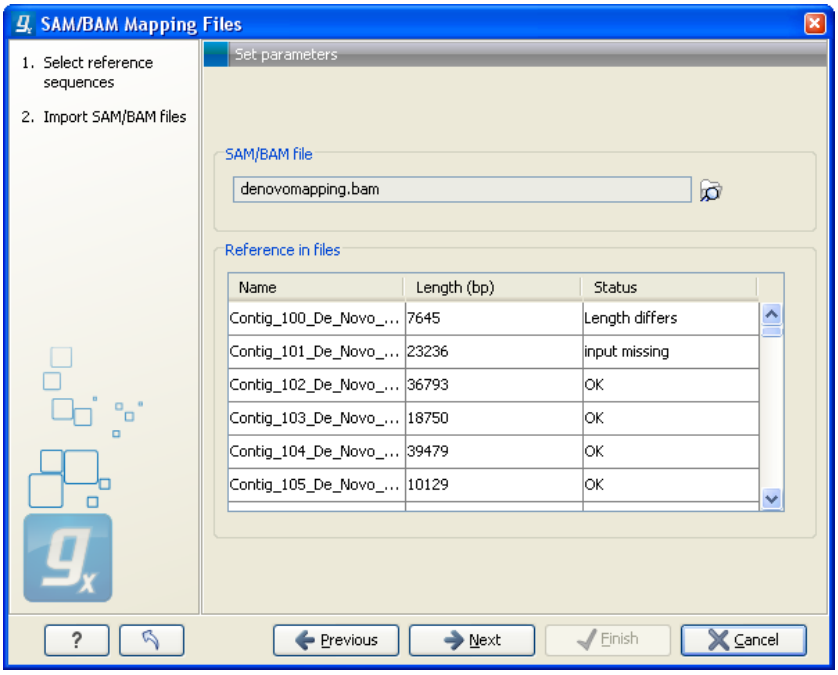
Reference in files

Name	Length (bp)	Status
Contig_100_De_Novo_...	7645	OK
Contig_101_De_Novo_...	23236	OK
Contig_102_De_Novo_...	36793	OK
Contig_103_De_Novo_...	18750	OK
Contig_104_De_Novo_...	39479	OK
Contig_105_De_Novo_...	10129	OK

Figure 19.11: *Selecting the SAM/BAM file containing all the read information.*

sequences selected from the Workbench. This can be used to double-check that the naming of the references are the same. (Note that reference sequences in a SAM/BAM file cannot contain spaces. If a reference sequence in the Workbench contains spaces, the space will be replaced with _ when comparing with the SAM/BAM file.). Figure19.12 shows an example where a reference sequence has not been provided (**input missing**) and one where the lengths of the reference sequences do not match (**Length differs**).

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save**



1. Select reference sequences
2. Import SAM/BAM files

Set parameters

SAM/BAM file: denovomapping.bam

Reference in files

Name	Length (bp)	Status
Contig_100_De_Novo_...	7645	Length differs
Contig_101_De_Novo_...	23236	input missing
Contig_102_De_Novo_...	36793	OK
Contig_103_De_Novo_...	18750	OK
Contig_104_De_Novo_...	39479	OK
Contig_105_De_Novo_...	10129	OK

Figure 19.12: *When there is inconsistency in the naming and sizes of reference sequences, this is shown in the dialog prior to import.*

in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis.

Note that this import operation is very memory-consuming for large data sets.

19.1.8 Tabular mapping files

The *CLC Genomics Workbench* supports import and export of files in tabular format such as Eland files coming from the Illumina Pipeline. The importer is quite flexible which means that it can be used to import any kind of mapping file in a tab-delimited format where each line in the file represents one read.

The idea behind the importer is that you import the mapping file which includes all the reads and then you specify one or more reference sequences which have already been imported into the Workbench. The Workbench will then combine the two to create mapping results () or mapping tables (). To import a tabular mapping file:

File | Import High-Throughput Sequencing Data () | Tabular Mapping Files ()

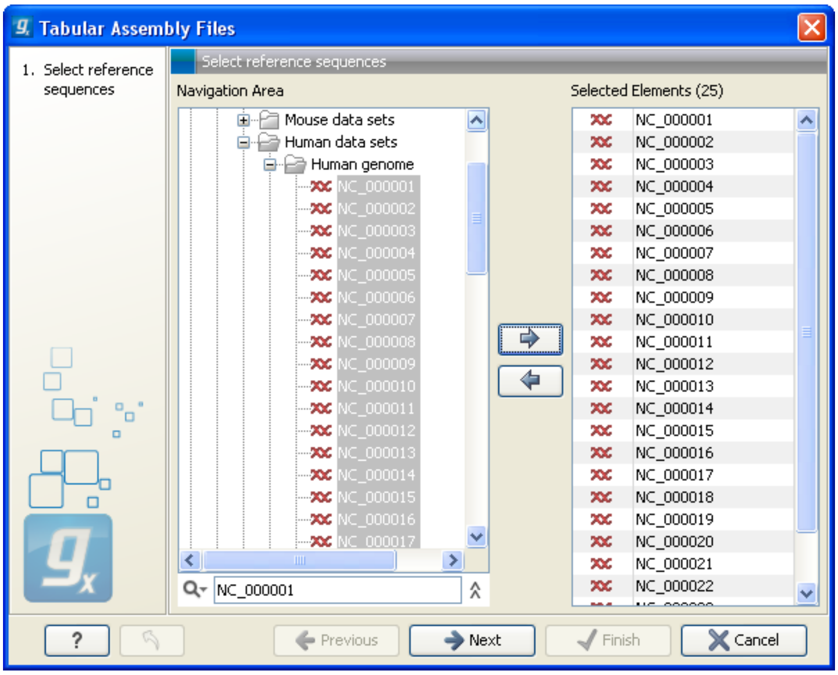
This will open a dialog where you choose the reference sequences to be used as shown in figure 19.13.

Select one or more reference sequence. Note that the name of your reference sequence has to match the reference name specified in the file. Click **Next**.

In this dialog, select () one or more tab delimited files as shown in figure 19.14.

Once the tab delimited file has been selected, you have to specify the following information:

- **Data columns**. The Workbench needs to know how the file is organized in order to create a



Tabular Assembly Files

1. Select reference sequences

Select reference sequences

Navigation Area

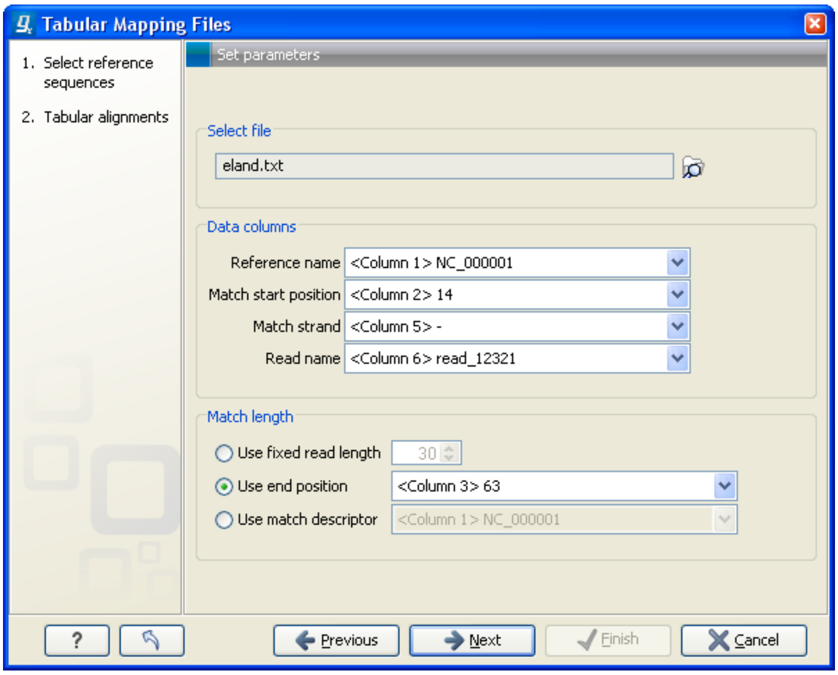
- Mouse data sets
- Human data sets
  - Human genome
    - NC_000001
    - NC_000002
    - NC_000003
    - NC_000004
    - NC_000005
    - NC_000006
    - NC_000007
    - NC_000008
    - NC_000009
    - NC_000010
    - NC_000011
    - NC_000012
    - NC_000013
    - NC_000014
    - NC_000015
    - NC_000016
    - NC_000017

NC_000001

Selected Elements (25)

- NC_000001
- NC_000002
- NC_000003
- NC_000004
- NC_000005
- NC_000006
- NC_000007
- NC_000008
- NC_000009
- NC_000010
- NC_000011
- NC_000012
- NC_000013
- NC_000014
- NC_000015
- NC_000016
- NC_000017
- NC_000018
- NC_000019
- NC_000020
- NC_000021
- NC_000022

Figure 19.13: *Defining reference sequences.*



Tabular Mapping Files

1. Select reference sequences
2. Tabular alignments

Set parameters

Select file

eland.txt

Data columns

Reference name <Column 1> NC_000001

Match start position <Column 2> 14

Match strand <Column 5> -

Read name <Column 6> read_12321

Match length

○ Use fixed read length 30

● Use end position <Column 3> 63

○ Use match descriptor <Column 1> NC_000001

Figure 19.14: *Defining reference sequences.*

result where the reads have been mapped correctly.

- **Reference name**. Select the column where the name reference sequence is specified. In the example above, this is in column 1.
- **Match start position**. The position on the reference sequence where the read is mapped. The numbering starts from position 1.
- **Match strand**. Whether the read is mapped the positive or negative strand. This should be specified using `F` / `R` (denoting forward and reverse reads) or `+` / `-`.

  - **Read name**. Whether the read is mapped the positive or negative strand. This should be specified using `F` / `R` (denoting forward and reverse reads) or `+` / `-`.

- **Match length**. The start position of the read is set above. In this section you specify the length of the match which can be done in any of the following ways:

  - **Use fixed read length**. If all reads have the same length, and if the read length or match end position is not provided in the file, you can specify a fixed length for all the reads.
  - **Use end position**. If you have a match end position just as a match start position, this can be used to determine match length.
  - **Use match descriptor**. This can be used to denote mismatches in the alignment. For a 35 base read, `35` denotes an exact match and `32C2` denotes substitution of a `C` at the 33rd position.

Note that the Workbench looks in the first line of the file to provide a preview when filling in this information.

Click **Next** to adjust how to handle the results (see section 9.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis.

Note that this import operation is very memory-consuming for large data sets.

19.2 Multiplexing

When you do batch sequencing of different samples, you can use multiplexing techniques to run different samples in the same run. There is often a data analysis challenge to separate the sequencing reads, so that the reads from one sample are mapped together. The *CLC Genomics Workbench* supports automatic grouping of samples for two multiplexing techniques:

- **By name**. This supports grouping of reads based on their name.
- **By sequence tag**. This supports grouping of reads based on information within the sequence (tagged sequences).

The details of these two functionalities are described below.

19.2.1 Sort sequences by name

With this functionality you will be able to group sequencing reads based on their file name. A typical example would be that you have a list of files named like this:

```
...
A02__Asp_F_016_2007-01-10
A02__Asp_R_016_2007-01-10
A02__Gln_F_016_2007-01-11
A02__Gln_R_016_2007-01-11
A03__Asp_F_031_2007-01-10
```

```
A03__Asp_R_031_2007-01-10
A03__Gln_F_031_2007-01-11
A03__Gln_R_031_2007-01-11
...
```

In this example, the names have five distinct parts (we take the first name as an example):

- **A02** which is the position on the 96-well plate
- **Asp** which is the name of the gene being sequenced
- **F** which describes the orientation of the read (forward/reverse)
- **016** which is an ID identifying the sample
- **2007-01-10** which is the data of the sequencing run

To start mapping these data, you probably want to have them divided into groups instead of having all reads in one folder. If, for example, you wish to map each sample separately, or if you wish to map each gene separately, you cannot simply run the mapping on all the sequences in one step.

That is where **Sort Sequences by Name** comes into play. It will allow you to specify which part of the name should be used to divide the sequences into groups. We will use the example described above to show how it works:

> **Toolbox | High-throughput Sequencing ( ) | Multiplexing ( ) | Sort Sequences by Name ( )**

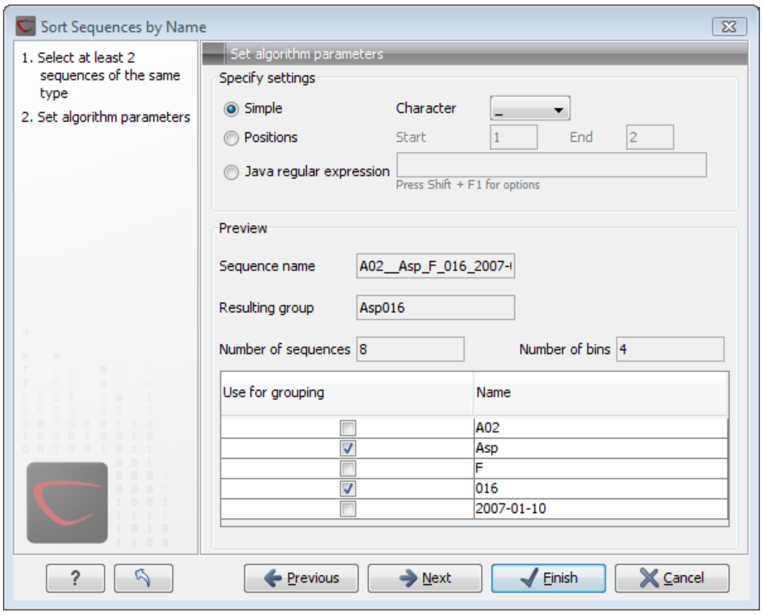
This opens a dialog where you can add the sequences you wish to sort. You can also add sequence lists or the contents of an entire folder by right-clicking the folder and choose: **Add folder contents**.

When you click **Next**, you will be able to specify the details of how the grouping should be performed. First, you have to choose how each part of the name should be identified. There are three options:

- **Simple**. This will simply use a designated character to split up the name. You can choose a character from the list:
  - Underscore _
  - Dash -
  - Hash (number sign / pound sign) #
  - Pipe |
  - Tilde ~
  - Dot .
- **Positions**. You can define a part of the name by entering the start and end positions, e.g. from character number 6 to 14. For this to work, the names have to be of equal lengths.
- **Java regular expression**. This is an option for advanced users where you can use a special syntax to have total control over the splitting. See more below.

In the example above, it would be sufficient to use a simple split with the underscore _ character, since this is how the different parts of the name are divided.

When you have chosen a way to divide the name, the parts of the name will be listed in the table at the bottom of the dialog. There is a checkbox next to each part of the name. This checkbox is used to specify which of the name parts should be used for grouping. In the example above, if we want to group the reads according to sample ID and gene name, these two parts should be checked as shown in figure 19.15.



Sort Sequences by Name

1. Select at least 2 sequences of the same type
2. Set algorithm parameters

Set algorithm parameters

Specify settings

● Simple Character _

○ Positions Start 1 End 2

○ Java regular expression

Press Shift + F1 for options

Preview

Sequence name A02__Asp_F_016_2007-(

Resulting group Asp016

Number of sequences 8 Number of bins 4

Use for grouping	Name
□	A02
☑	Asp
□	F
☑	016
□	2007-01-10

Figure 19.15: *Splitting up the name at every underscore (_) and using the sample ID and gene name for grouping.*

At the middle of the dialog there is a preview panel listing:

- **Sequence name**. This is the name of the first sequence that has been chosen. It is shown here in the dialog in order to give you a sample of what the names in the list look like.
- **Resulting group**. The name of the group that this sequence would belong to if you proceed with the current settings.
- **Number of sequences**. The number of sequences chosen in the first step.
- **Number of groups**. The number of groups that would be produced when you proceed with the current settings.

This preview cannot be changed. It is shown to guide you when finding the appropriate settings.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. A new sequence list will be generated for each group. It will be named according to the group, e.g. *Asp016* will be the name of one of the groups in the example shown in figure 19.15.

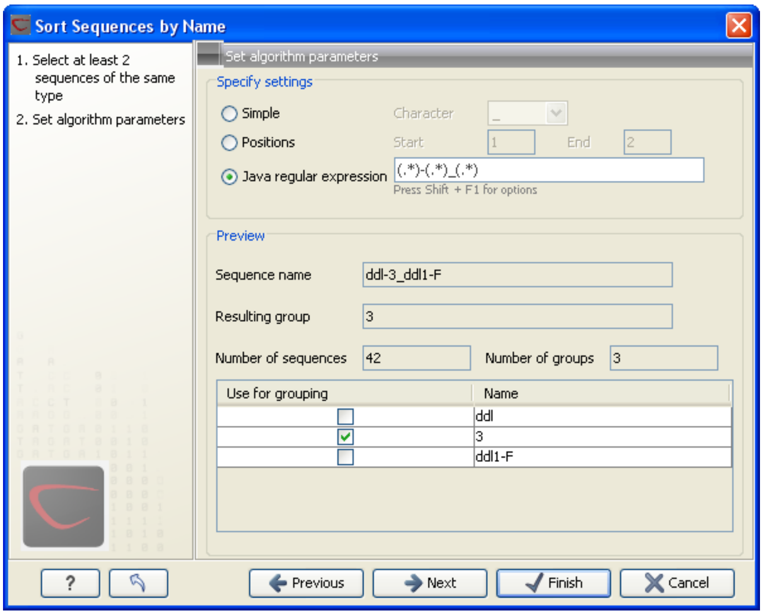
Advanced splitting using regular expressions

You can see a more detail explanation of the regular expressions syntax in section 14.7.3. In this section you will see a practical example showing how to create a regular expression. Consider a list of files as shown below:

```
...
adk-29_adk1n-F
adk-29_adk2n-R
adk-3_adk1n-F
adk-3_adk2n-R
adk-66_adk1n-F
adk-66_adk2n-R
atp-29_atpA1n-F
atp-29_atpA2n-R
atp-3_atpA1n-F
atp-3_atpA2n-R
atp-66_atpA1n-F
atp-66_atpA2n-R
...
```

In this example, we wish to group the sequences into three groups based on the number after the "-" and before the "_" (i.e. 29, 3 and 66). The simple splitting as shown in figure 19.15 requires the same character before and after the text used for grouping, and since we now have both a "-" and a "_", we need to use the regular expressions instead (note that dividing by position would not work because we have both single and double digit numbers (3, 29 and 66)).

The regular expression for doing this would be `(.*)-(.*)_(.*)` as shown in figure 19.16.



1. Select at least 2 sequences of the same type
2. Set algorithm parameters

Set algorithm parameters

Specify settings

○ Simple Character _

○ Positions Start 1 End 2

● Java regular expression (.*)-(.*)_(.*)

Press Shift + F1 for options

Preview

Sequence name: ddl-3_ddl1-F

Resulting group: 3

Number of sequences: 42 Number of groups: 3

Use for grouping	Name
□	ddl
☑	3
□	ddl1-F

Figure 19.16: *Dividing the sequence into three groups based on the number in the middle of the name.*

The round brackets () denote the part of the name that will be listed in the groups table at the bottom of the dialog. In this example we actually did not need the first and last set of brackets, so the expression could also have been `.*-(.*)_.*` in which case only one group would be listed in the table at the bottom of the dialog.

19.2.2 Process tagged sequences

Multiplexing as described in section 19.2.1 is of course only possible if proper sequence names could be assigned from the sequencing process. With many of the new high-throughput technologies, this is not possible.

However, there is a need for being able to input several different samples to the same sequencing run, so multiplexing is still relevant - it just has to be based on another way of identifying the sequences. A method has been proposed to *tag* the sequences with a unique identifier during the preparation of the sample for sequencing [Meyer et al., 2007].

With this technique, each sequence will have a sample-specific tag - a special sequence of nucleotides before and after the sequence of interest. This principle is shown in figure 19.17 (please refer to [Meyer et al., 2007] for more detailed information).

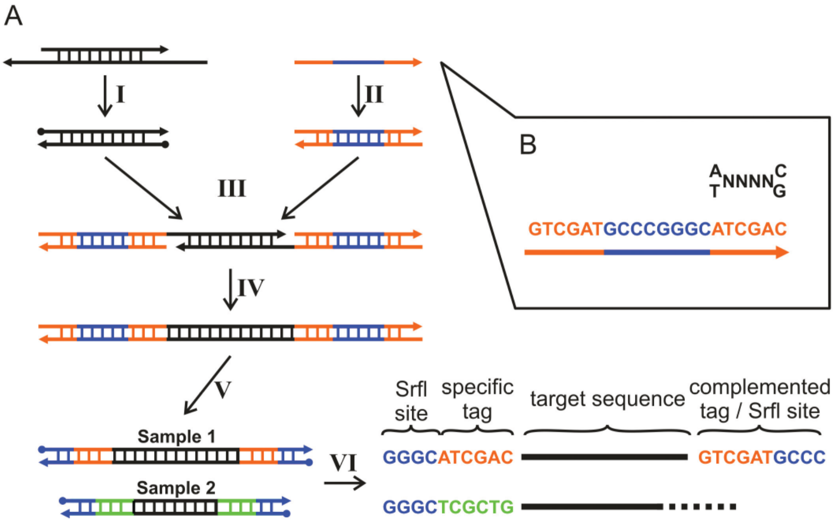


Figure 19.17: *Tagging the target sequence. Figure from [Meyer et al., 2007].*

The sample-specific tag - also called the barcode - can then be used to distinguish between the different samples when analyzing the sequence data. This post-processing of the sequencing data has been made easy by the multiplexing functionality of the *CLC Genomics Workbench* which simply divides the data into separate groups prior to analysis. Note that there is also an example using Illumina data at the end of this section.

Before processing the data, you need to import it as described in section 19.1.

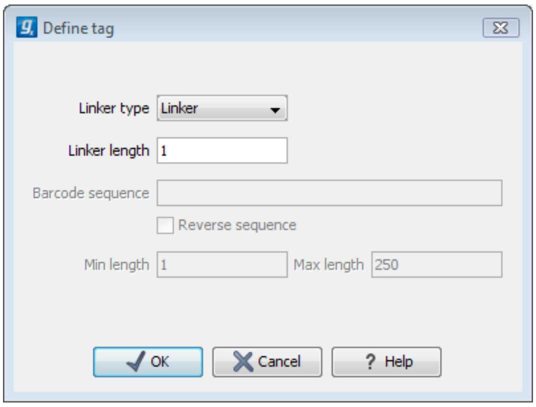
The first step is to separate the imported sequence list into sublists based on the barcode of the sequences:

> **Toolbox | High-throughput Sequencing ( ) | Multiplexing ( ) | Process Tagged Sequences ( )**

This opens a dialog where you can add the sequences you wish to sort. You can also add sequence lists.

When you click **Next**, you will be able to specify the details of how the de-multiplexing should be performed. At the bottom of the dialog, there are three buttons which are used to **Add**, **Edit** and **Delete** the elements that describe how the barcode is embedded in the sequences.

First, click **Add** to define the first element. This will bring up the dialog shown in 19.18.



Linker type: Linker

Linker length: 1

Barcode sequence:

☐ Reverse sequence

Min length: 1 Max length: 250

Figure 19.18: *Defining an element of the barcode system.*

At the top of the dialog, you can choose which kind of element you wish to define:

- **Linker**. This is a sequence which should just be ignored - it is neither the barcode nor the sequence of interest. Following the example in figure 19.17, it would be the four nucleotides of the *SrfI* site. For this element, you simply define its length - nothing else.
- **Barcode**. The barcode is the stretch of nucleotides used to group the sequences. For that, you need to define what the valid bases are. This is done when you click **Next**. In this dialog, you simply need to specify the length of the barcode.
- **Sequence**. This element defines the sequence of interest. You can define a length interval for how long you expect this sequence to be. The sequence part is the only part of the read that is retained in the output. Both barcodes and linkers are removed.

The concept when adding elements is that you add e.g. a linker, a barcode and a sequence in the desired sequential order to describe the structure of each sequencing read. You can of course edit and delete elements by selecting them and clicking the buttons below. For the example from figure 19.17, the dialog should include a linker for the *SrfI* site, a barcode, a sequence, a barcode (now reversed) and finally a linker again as shown in figure 19.19.

If you have paired data, the dialog shown in figure 19.19 will be displayed twice - one for each part of the pair.

Clicking **Next** will display a dialog as shown in figure 19.20.

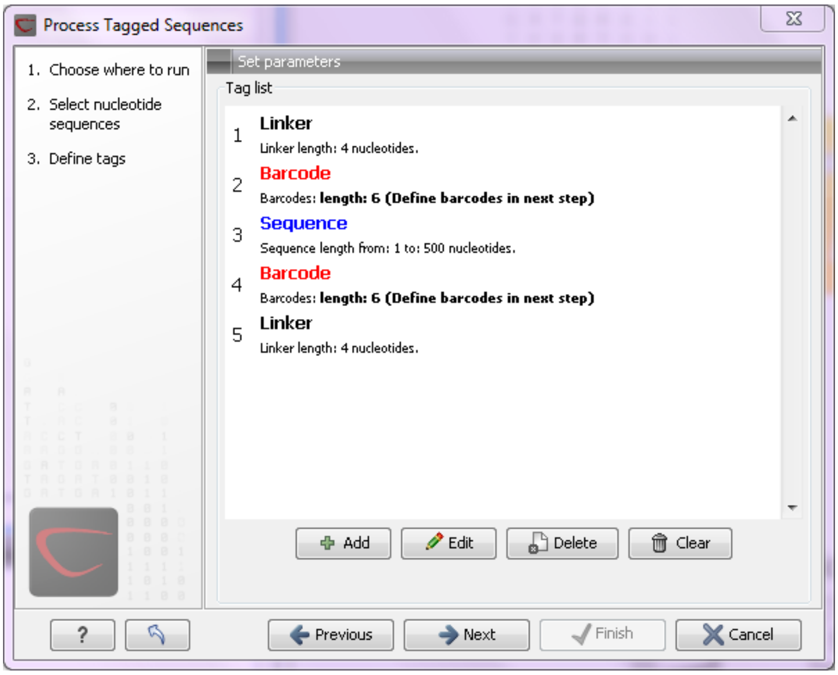
The barcodes can be entered manually by clicking the **Add** (➡) button. You can edit the barcodes and the names by clicking the cells in the table. The name is used for naming the results.

In addition to adding barcodes manually, you can also **Import** (📂) barcode definitions from an Excel or CSV file. The format of the file needs to be like this:

ATCGAC	ATCGAC	Sample 1
TCGCTG	TCGCTG	Sample 2

If you want to check the formatting, you can also add a barcode manually and **Export** (📂) the barcode definition to see what it looks like.

The **Preview** column will show a preview of the results by running through the first 10,000 reads. Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.



1 **Linker**
Linker length: 4 nucleotides.

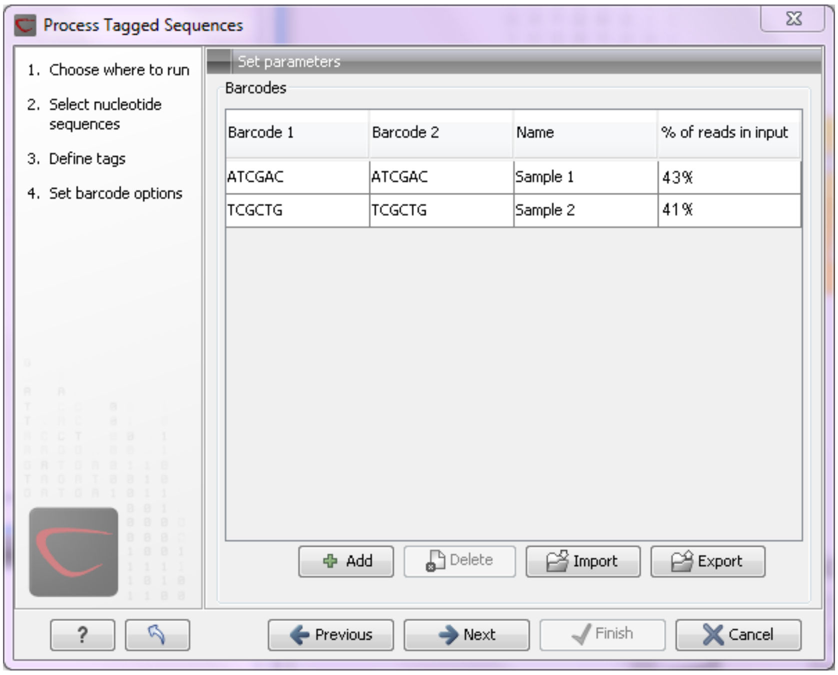
2 **Barcode**
Barcodes: **length: 6 (Define barcodes in next step)**

3 **Sequence**
Sequence length from: 1 to: 500 nucleotides.

4 **Barcode**
Barcodes: **length: 6 (Define barcodes in next step)**

5 **Linker**
Linker length: 4 nucleotides.

Figure 19.19: *Processing the tags as shown in the example of figure 19.17.*



Barcode 1	Barcode 2	Name	% of reads in input
ATCGAC	ATCGAC	Sample 1	43%
TCGCTG	TCGCTG	Sample 2	41%

Figure 19.20: *Specifying the barcodes as shown in the example of figure 19.17.*

Note that there is also an option to create subfolders for each sequence list. This can be handy when the results need to be processed in batch mode (see section 9.1).

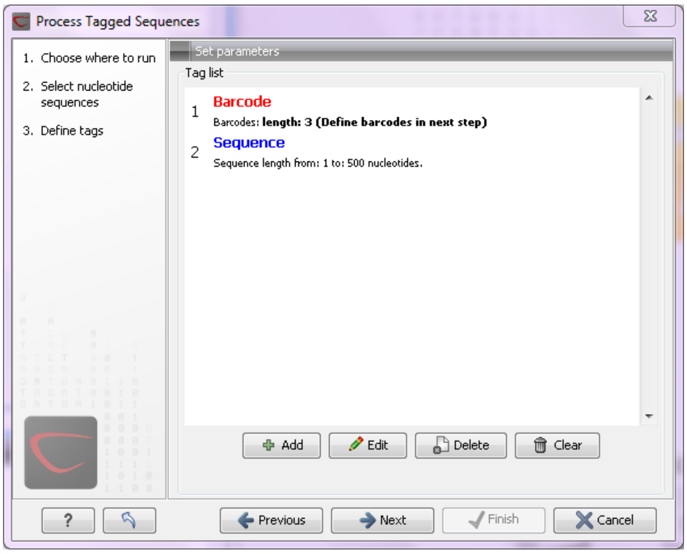
A new sequence list will be generated for each barcode containing all the sequences where this barcode is identified. Both the linker and barcode sequences are removed from each of the sequences in the list, so that only the target sequence remains. This means that you can continue the analysis by doing trimming or mapping. Note that you have to perform separate mappings for each sequence list.

An example using Illumina barcoded sequences

The data set in this example can be downloaded from the Short Read Archive at NCBI: `http://www.ncbi.nlm.nih.gov/sites/entrez?db=sra&term=SRX014012`.

The barcoding was done using the following tags at the beginning of each read: CCT, AAT, GGT, CGT (see supplementary material of [Cronn et al., 2008] at `http://nar.oxfordjournals.org/cgi/data/gkn502/DC1/1`).

The settings in the dialog should thus be as shown in figure 19.21.



Process Tagged Sequences

1. Choose where to run
2. Select nucleotide sequences
3. Define tags

Set parameters

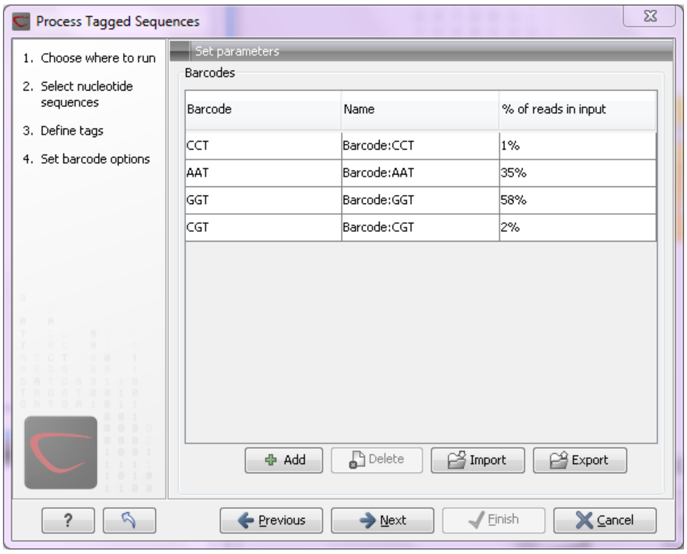
Tag list

1 Barcode
Barcodes: length: 3 (Define barcodes in next step)

2 Sequence
Sequence length from: 1 to: 500 nucleotides.

Figure 19.21: *Setting the barcode length at three*

Click **Next** to specify the bar codes as shown in figure 19.22 (use the **Add** button).



Process Tagged Sequences

1. Choose where to run
2. Select nucleotide sequences
3. Define tags
4. Set barcode options

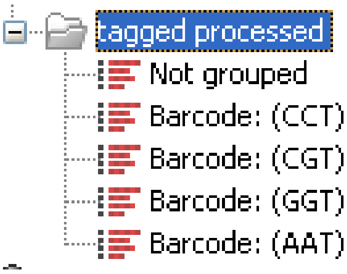
Set parameters

Barcodes

Barcode	Name	% of reads in input
CCT	Barcode:CCT	1%
AAT	Barcode:AAT	35%
GGT	Barcode:GGT	58%
CGT	Barcode:CGT	2%

Figure 19.22: *A preview of the result*

With this data set we got the four groups as expected (shown in figure 19.23). The **Not grouped** list contains 445,560 reads that will have to be discarded since they do not have any of the barcodes.


tagged processed

Not grouped

Barcode: (CCT)

Barcode: (CGT)

Barcode: (GGT)

Barcode: (AAT)

Figure 19.23: *The result is one sequence list per barcode and a list with the remainders*

19.3 Trim sequences

CLC Genomics Workbench offers a number of ways to trim your sequence reads prior to assembly and mapping, including adapter trimming, quality trimming and length trimming. Note that different types of trimming are performed sequentially in the same order as they appear in the trim dialogs:

1. Quality trimming based on quality scores
2. Ambiguity trimming to trim off e.g. stretches of Ns
3. Adapter trimming
4. Base trim to remove a specified number of bases at either 3' or 5' end of the reads
5. Length trimming to remove reads shorter or longer than a specified threshold

The result of the trim is a list of sequences that have passed the trim (referred to as the trimmed list below) and optionally a list of the sequences that have been discarded and a summary report (list of discarded sequences). The original data will be not be changed.

To start trimming:

Toolbox | **High-throughput Sequencing (**▶**)** | **Trim Sequences (**✂**)**

This opens a dialog where you can add sequences or sequence lists. If you add several sequence lists, each list will be processed separately and you will get a a list of trimmed sequences for each input sequence list.

When the sequences are selected, click **Next**.

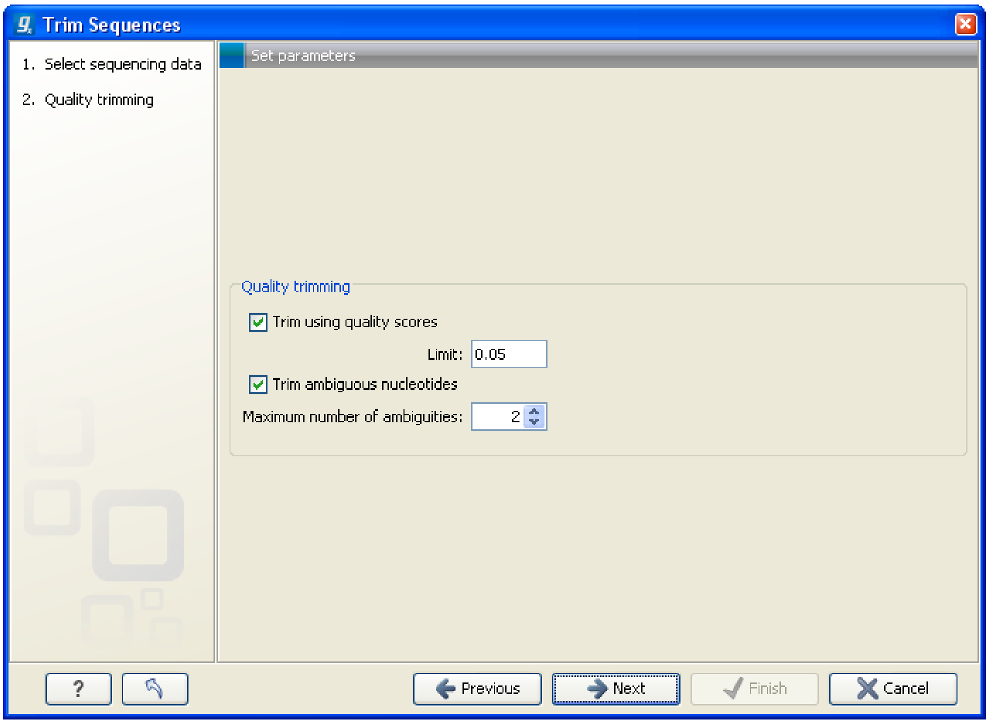
19.3.1 Quality trimming

This opens the dialog displayed in figure 19.24 where you can specify parameters for quality trimming.

The following parameters can be adjusted in the dialog:

- **Trim using quality scores.** If the sequence files contain quality scores from a base-caller algorithm this information can be used for trimming sequence ends. The program uses the modified-Mott trimming algorithm for this purpose (Richard Mott, personal communication):

  Quality scores in the Workbench are on a Phred scale in the Workbench (formats using other scales are converted during import). First step in the trim process is to convert the



Quality trimming

☑ Trim using quality scores

Limit: 0.05

☑ Trim ambiguous nucleotides

Maximum number of ambiguities: 2

Figure 19.24: *Specifying quality trimming.*

quality score (Q) to error probability: $p_{error} = 10^{\frac{Q}{-10}}$. (This now means that low values are high quality bases.)

Next, for every base a new value is calculated: $Limit - p_{error}$. This value will be negative for low quality bases, where the error probability is high.

For every base, the Workbench calculates the running sum of this value. If the sum drops below zero, it is set to zero. The part of the sequence to be retained after trimming is the region between the first positive value of the running sum and the highest value of the running sum. Everything before and after this region will be trimmed off.

A read will be completely removed if the score never makes it above zero.

At http://www.clcbio.com/files/usermanuals/trim.zip you find an example sequence and an Excel sheet showing the calculations done for this particular sequence to illustrate the procedure described above.

- **Trim ambiguous nucleotides.** This option trims the sequence ends based on the presence of ambiguous nucleotides (typically N). Note that the automated sequencer generating the data must be set to output ambiguous nucleotides in order for this option to apply. The algorithm takes as input the *maximal number of ambiguous nucleotides allowed in the sequence after trimming*. If this maximum is set to e.g. 3, the algorithm finds the maximum length region containing 3 or fewer ambiguities and then trims away the ends not included in this region.

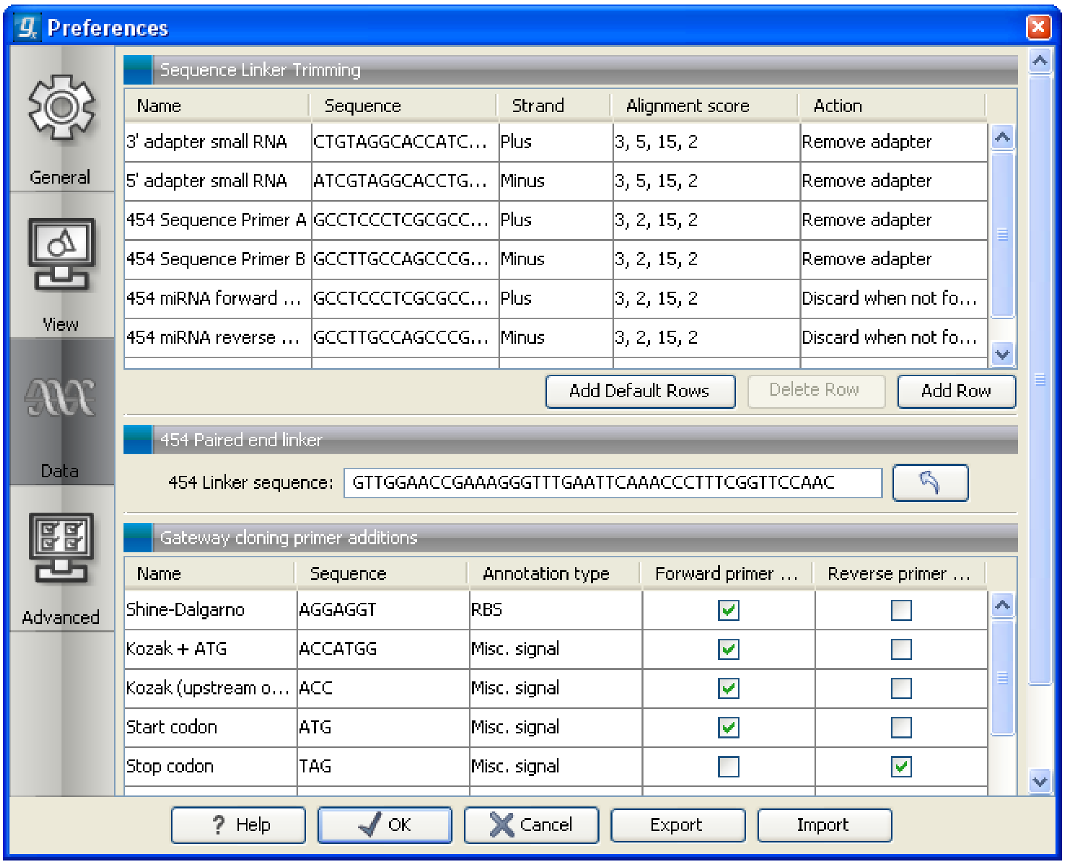
19.3.2 Adapter trimming

Clicking **Next** will allow you to specify adapter trimming.

The *CLC Genomics Workbench* comes with a set of predefined adapter sequences from the most common kits provided by the high-throughput sequencing vendors. You can easily add or modify the adapters on this list in the preferences:

Edit | **Preferences (** ⚙ **)** | **Data**

This will display the adapter trim panel as shown in figure 19.25 where each row represents an adapter sequence including the settings used for trimming.



Sequence Linker Trimming

Name	Sequence	Strand	Alignment score	Action
3' adapter small RNA	CTGTAGGCACCATC...	Plus	3, 5, 15, 2	Remove adapter
5' adapter small RNA	ATCGTAGGCACCTG...	Minus	3, 5, 15, 2	Remove adapter
454 Sequence Primer A	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Remove adapter
454 Sequence Primer B	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Remove adapter
454 miRNA forward ...	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Discard when not fo...
454 miRNA reverse ...	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Discard when not fo...

454 Paired end linker

454 Linker sequence: GTTGGAACCGAAAGGGTTTGAATTCAAACCCTTTCGGTTCCAAC

Gateway cloning primer additions

Name	Sequence	Annotation type	Forward primer ...	Reverse primer ...
Shine-Dalgarno	AGGAGGT	RBS	☑	□
Kozak + ATG	ACCATGG	Misc. signal	☑	□
Kozak (upstream o...	ACC	Misc. signal	☑	□
Start codon	ATG	Misc. signal	☑	□
Stop codon	TAG	Misc. signal	□	☑

Figure 19.25: *Editing the set of adapters for adapter trimming.*

At the bottom of the panel, you have the following options:

- **Add Default Rows.** If you have deleted or changed the pre-defined set of adapters, you can add them to the list using this button (note that they will not replace existing adapters).
- **Delete Row**. Delete the selected adapter.
- **Add Row**. Add a new empty row where you can specify your own adapter settings.

All the information in the panel can be edited by clicking or double-clicking. The **Strand**, **Alignment score** and **Action** settings can also be modified when running the trim (see figure 19.31).

Action to perform when a match is found

For each read sequence in the input to trim, the Workbench performs a Smith-Waterman alignment [Smith and Waterman, 1981] with the adapter sequence to see if there is a match (details described below). When a match is found, the user can specify three kinds of actions:

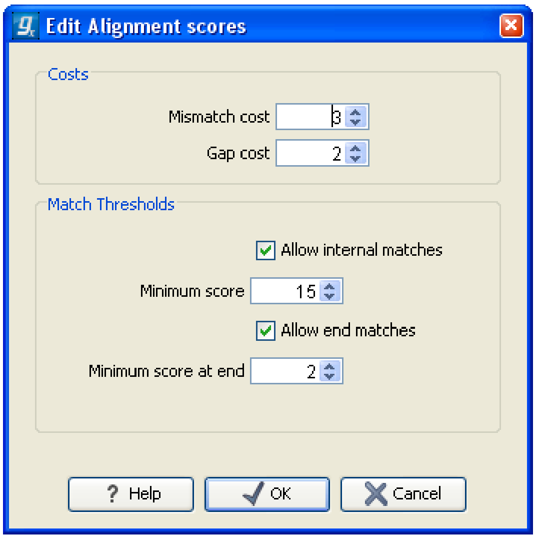
- **Remove adapter.** This will remove the adapter *and all the nucleotides 5' of the match*. All the nucleotides 3' of the adapter match will be preserved in the read that will be retained in the trimmed reads list. If there are no nucleotides 3' of the adapter match, the read is added to the **List of discarded sequences** (see section 19.3.4).

- **Discard when not found**. If a match is found, the adapter sequence is removed (including all nucleotides 5' of the match as described above) and the rest of the sequence is retained in the list of trimmed reads. If no match is found, the whole sequence is discarded and put in the list of discarded sequences. This kind of adapter trimming is useful for small RNA sequencing where the remnants of the adapter is an indication that this is indeed a small RNA.

- **Discard when found**. If a match is found, the read is discarded. If no match is found, the read is retained in the list of trimmed reads. This can be used for quality checking the data for linker contaminations etc.

When is there a match?

To determine whether there is a match there is a set of scoring thresholds that can be adjusted and inspected by double-clicking the **Alignment score** column. This will bring up a dialog as shown in figure 19.26.



Edit Alignment scores

Costs

Mismatch cost: 3

Gap cost: 2

Match Thresholds

☑ Allow internal matches

Minimum score: 15

☑ Allow end matches

Minimum score at end: 2

Figure 19.26: *Setting the scoring thresholds for adapter trimming.*

At the top, you can choose the costs for mismatch and gaps. A match is rewarded one point (this cannot be changed), and per default a mismatch costs 2 and a gap (insertion or deletion) costs 3. A few examples of adapter matches and corresponding scores are shown below.

In the panel below, you can set the **Minimum score** for a match to be accepted. Note that there is a difference between an **internal match** and an **end match**. The examples above are all internal matches where the alignment of the adapter falls within the read. Below are a few examples showing an adapter match at the end:

In the first two examples, the adapter sequence extends beyond the end of the read. This is what typically happens when sequencing e.g. small RNAs where you sequence part of the adapter. The third example shows an example which could be interpreted both as an end match and an

```
    CGTATCAATCGATTACGCTATGAATG
a)      ||||||| ||||            11 matches - 2 mismatches = 7
       TTCAATCGGTTAC


    CGTATCAATCGATTACGCTATGAATG
       |||||||||| ||||          14 matches - 1 gap = 11
b)     ATCAATCGAT-CGCT


    CGTATCAATCGATTACGCTATGAATG
c)      |||||||                 7 matches - 3 mismatches = 1
       TTCAATCGGG
```

Figure 19.27: Three examples showing a sequencing read (top) and an adapter (bottom). The examples are artificial, using default setting with mismatch costs = 2 and gap cost = 3.

```
        CGTATCAATCGATTACGCTATGAATG
d)      |||||                         5 matches = 5 (as end match)
    GATTCGTAT


        CGTATCAATCGATTACGCTATGAATG
e)      || ||||                       6 matches - 1 mismatch = 4 (as end match)
    GATTCGCATCA


    CGTATCAATCGATTACGCTATGAATG
f)  |||| |||||                        9 matches - 1 gap = 6 (as end match)
    CGTA-CAATC


    CGTATCAATCGATTACGCTATGAATG
g)                  ||||||||||        10 matches = 10 (as internal match)
                    GCTATGAATG
```

Figure 19.28: Four examples showing a sequencing read (top) and an adapter (bottom). The examples are artificial.

internal match. However, the Workbench will interpret this as an end match, because it starts at beginning (5' end) of the read. Thus, the definition of an end match is that the alignment of the adapter starts at the read's 5' end. The last example could also be interpreted as an end match, but because it is a the 3' end of the read, it counts as an internal match (this is because you would not typically expect partial adapters at the 3' end of a read). Also note, that if **Remove adapter** is chosen for the last example, the full read will be discarded because everything 5' of the adapter is removed.

Below, the same examples are re-iterated showing the results when applying different scoring schemes. In the first round, the settings are:

- Allowing internal matches with a minimum score of 6
- Not allowing end matches
- Action: Remove adapter

The result would be the following (the retained parts are green):

```
    CGTATCAATCGATTACGCTATGAATG
a)      ||||||| ||||                 11 matches - 2 mismatches = 7
       TTCAATCGGTTAC


    CGTATCAATCGATTACGCTATGAATG
       |||||||||| ||||               14 matches - 1 gap = 11
b)     ATCAATCGAT-CGCT


    CGTATCAATCGATTACGCTATGAATG
c)      |||||||                       7 matches - 3 mismatches = 1
       TTCAATCGGG


        CGTATCAATCGATTACGCTATGAATG
d)      |||||                         5 matches = 5 (as end match)
    GATTCGTAT


        CGTATCAATCGATTACGCTATGAATG
e)      || ||||                       6 matches - 1 mismatch = 4 (as end match)
    GATTCGCATCA


    CGTATCAATCGATTACGCTATGAATG
f)  |||| |||||                        9 matches - 1 gap = 6 (as end match)
    CGTA-CAATC


    CGTATCAATCGATTACGCTATGAATG
g)                  ||||||||||       10 matches = 10 (as internal match)
                    GCTATGAATG
```

Figure 19.29: The results of trimming with internal matches only. Red is the part that is removed and green is the retained part. Note that the read at the bottom is completely discarded.

A different set of adapter settings could be:

- Allowing internal matches with a minimum score of 11
- Allowing end match with a minimum score of 4
- Action: Remove adapter

The result would be:

Other adapter trimming options

When you run the trim, you specify the adapter settings as shown in figure 19.31.

You select an adapter to be used for trimming by checking the checkbox next to the adapter name. You can overwrite the settings defined in the preferences regarding **Strand**, **Alignment score** and **Action** by simply clicking or double-clicking in the table.

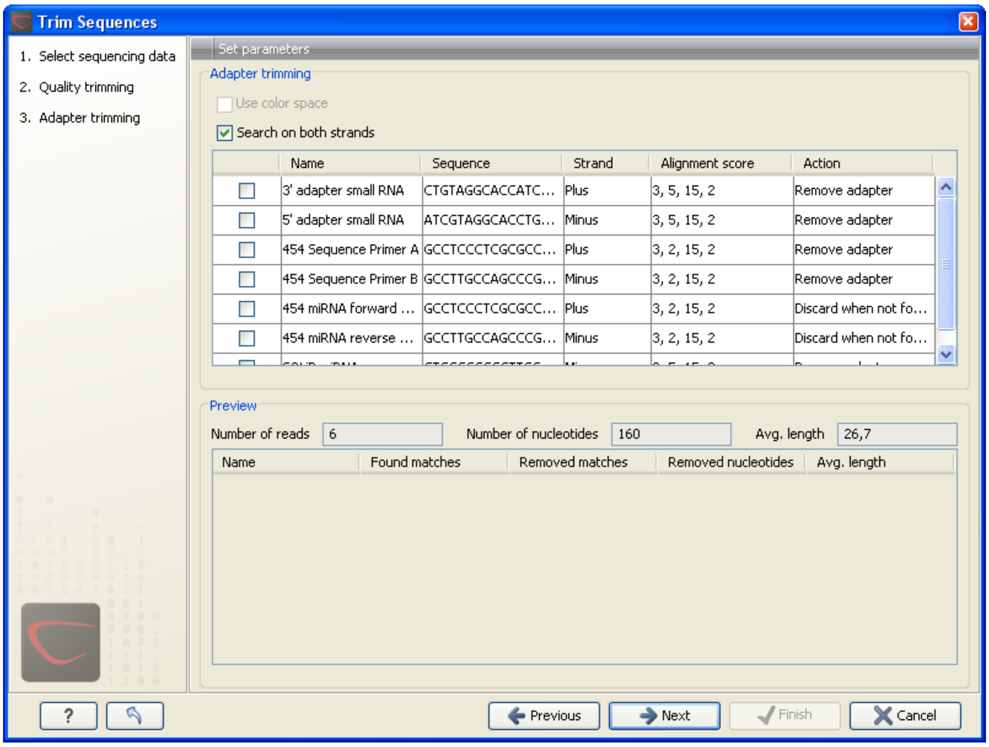
```
    CGTATCAATCGATTACGCTATGAATG
a)      ||||||| ||||                 11 matches - 2 mismatches = 7
       TTCAATCGGTTAC


    CGTATCAATCGATTACGCTATGAATG
       |||||||||| ||||               14 matches - 1 gap = 11
b)     ATCAATCGAT-CGCT


    CGTATCAATCGATTACGCTATGAATG
c)      |||||||                       7 matches - 3 mismatches = 1
       TTCAATCGGG


        CGTATCAATCGATTACGCTATGAATG
d)      |||||                         5 matches = 5 (as end match)
    GATTCGTAT


        CGTATCAATCGATTACGCTATGAATG
e)      || ||||                       6 matches - 1 mismatch = 4 (as end match)
    GATTCGCATCA


    CGTATCAATCGATTACGCTATGAATG
f)  |||| |||||                        9 matches - 1 gap = 6 (as end match)
    CGTA-CAATC


    CGTATCAATCGATTACGCTATGAATG
g)                  ||||||||||       10 matches = 10 (as internal match)
                    GCTATGAATG
```

Figure 19.30: The results of trimming with both internal and end matches. Red is the part that is removed and green is the retained part.

At the top, you can specify if the adapter trimming should be performed in **Color space**. Note that this option is only available for sequencing data imported using the SOLiD import (see section 19.1.3). When doing the trimming in color space, the Smith-Waterman alignment is simply done using colors rather than bases. The adapter sequence is still input in base space, and the Workbench then infers the color codes. Note that the scoring thresholds apply to the color space alignment (this means that a perfect match of 10 bases would get a score of 9 because 10 bases are represented by 9 color residues). Learn more about color space in section 19.8.

Besides defining the **Action** and **Alignment scores**, you can also define on which strand the adapter should be found. This can be done in two ways:

- Defining either **Plus** or **Minus** for the individual adapter sequence (this can be done either in the **Preferences** or in the dialog shown in figure 19.31). Note that all the definitions above regarding 3' end and 5' end also apply to the minus strand (i.e. selecting the Minus strand is equivalent to reverse complementing all the reads). The adapter in this case should be defined as you would see it on the plus strand of the reverse complemented read. Figure 19.32 below shows a few examples of an adapter defined on the minus strand.



□ Use color space

☑ Search on both strands

	Name	Sequence	Strand	Alignment score	Action
□	3' adapter small RNA	CTGTAGGCACCATC...	Plus	3, 5, 15, 2	Remove adapter
□	5' adapter small RNA	ATCGTAGGCACCTG...	Minus	3, 5, 15, 2	Remove adapter
□	454 Sequence Primer A	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Remove adapter
□	454 Sequence Primer B	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Remove adapter
□	454 miRNA forward ...	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Discard when not fo...
□	454 miRNA reverse ...	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Discard when not fo...

Number of reads: 6 Number of nucleotides: 160 Avg. length: 26,7

Name	Found matches	Removed matches	Removed nucleotides	Avg. length

Figure 19.31: *Trimming your sequencing data for adapter sequences.*

- Checking the **Search on both strands** checkbox will search both the minus and plus strand for the adapter sequence (the result would be equivalent to defining two adapters and searching one on the plus strand and one on the minus strand).

Below is an example showing hits for an adapter sequence defined as `CTGCTGTACGGCCAAGGCG`, searching on the minus strand. You can see that if you reverse complemented the adapter you

```
     ACCGAGAAACGCCTTGGCCGTACAGCAG
a)            |||||||||||||||||||     19 matches = 19
              ⅁Ɔ⅁⅁ⱯⱯƆƆ⅁⅁ƆⱯ⊥⅁⊥Ɔ⅁⊥Ɔ

     ACCGATAAACGCCTTGGCCGTACAGCAGATGCC
b)            ||||||||| |||||||||     18 matches - 2 mismatches = 16
              ⅁Ɔ⅁⅁ⱯⱯƆƆ⅁⅁ƆⱯ⊥⅁⊥Ɔ⅁⊥Ɔ
```

Figure 19.32: An adapter defined as CTGCTGTACGGCCAAGGCG searching on the minus strand. Red is the part that is removed and green is the retained part. The retained part is 3' of the match on the minus strand, just like matches on the plus strand.

would find the hit on the plus strand, but then you would have trimmed the wrong end of the read. So it is important to define the adapter as it is, without reverse complementing.

Below the adapter table you find a preview listing the results of trimming with the current settings on 1000 reads in the input file (reads 1001-2000 when the read file is long enough). This is useful for a quick feedback on how changes in the parameters affect the trimming (rather than having to run the full analysis several times to identify a good parameter set). The following information is shown:

- **Name**. The name of the adapter.

- **Matches found**. Number of matches found based on the strand and alignment score settings.
- **Reads discarded**. This is the number of reads that will be completely discarded. This can either be because they are completely trimmed (when the **Action** is set to Remove adapter and the match is found at the 3' end of the read), or when the **Action** is set to Discard when found or Discard when not found.
- **Nucleotides removed**. The number of nucleotides that are trimmed include both the ones coming from the reads that are discarded and the ones coming from the parts of the reads that are trimmed off.
- **Avg. length** This is the average length of the reads that are retained (excluding the ones that are discarded).

Note that the preview panel is only showing how the adapter trim affects the results. If other kinds of trimming (quality or length trimming) is applied, this will not be reflected in the preview but still influence the results.

Next time you run the trimming, your previous settings will automatically be remembered. Note that if you change settings in the **Preferences**, they may not be updated when running trim because the last settings are always used. Any conflicts are illustrated with text in *italics*. To make the updated preference take effect, press the **Reset to CLC Standard Settings** ( ) button.

19.3.3 Length trimming

Clicking **Next** will allow you to specify length trimming as shown in figure 19.33.

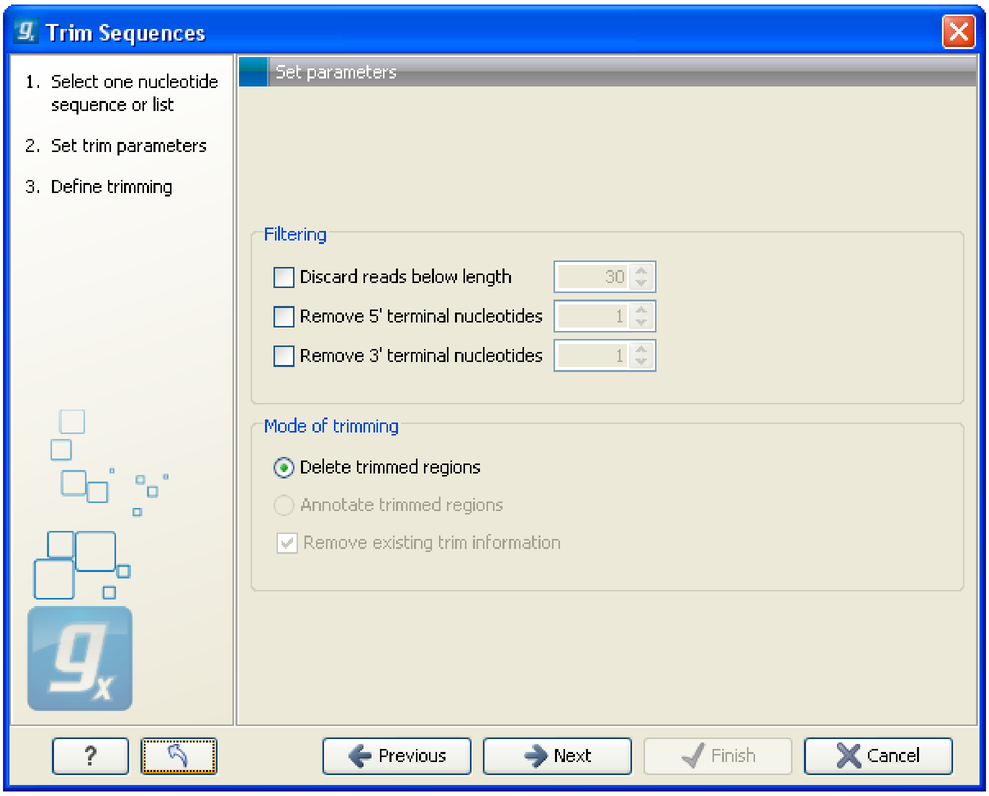
At the top you can choose to **Trim bases** by specifying a number of bases to be removed from either the 3' or the 5' end of the reads. Below you can choose to **Discard reads below length**. This can be used if you wish to simply discard reads because they are too short. Similarly, you can discard reads above a certain length. This will typically be useful when investigating e.g. small RNAs (note that this is an integral part of the small RNA analysis together with adapter trimming).

19.3.4 Trim output

Clicking **Next** will allow you to specify the output of the trimming as shown in figure 19.34.

No matter what is chosen here, the list of trimmed reads will always be produced. In addition the following can be output as well:

- **Create list of discarded sequences**. This will produce a list of reads that have been discarded during trimming. When only parts of the read has been discarded, it will now show up in this list.
- **Create report**. An example of a trim report is shown in figure 19.35. The report includes the following:
  - **Trim summary.**
    - **Name.** The name of the sequence list used as input.



1. Select one nucleotide sequence or list
2. Set trim parameters
3. Define trimming

Set parameters

Filtering

□ Discard reads below length 30

□ Remove 5' terminal nucleotides 1

□ Remove 3' terminal nucleotides 1

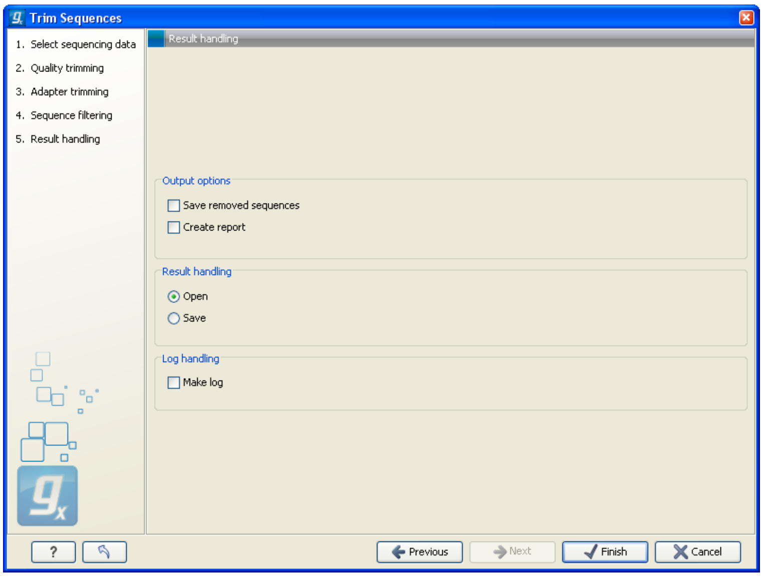
Mode of trimming

● Delete trimmed regions

○ Annotate trimmed regions

☑ Remove existing trim information

Figure 19.33: *Trimming on length.*



1. Select sequencing data
2. Quality trimming
3. Adapter trimming
4. Sequence filtering
5. Result handling

Result handling

Output options

□ Save removed sequences

□ Create report

Result handling

● Open

○ Save

Log handling

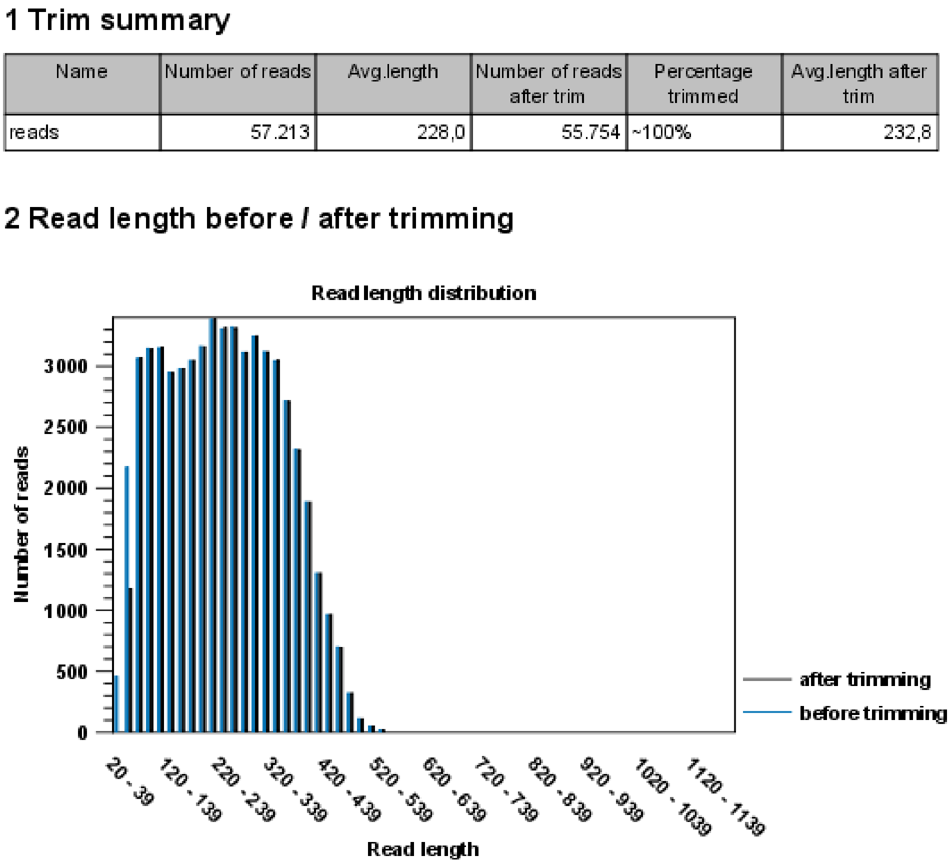
□ Make log

Figure 19.34: *Specifying the trim output. No matter what is chosen here, the list of trimmed reads will always be produced.*

- ∗ **Number of reads.** Number of reads in the input file.
- ∗ **Avg. length.** Average length of the reads in the input file.
- ∗ **Number of reads after trim.** The number of reads retained after trimming.
- ∗ **Percentage trimmed.** The percentage of the input reads that are retained.
- ∗ **Avg. length after trim.** The average length of the retained sequences.

– **Read length before / after trimming**. This is a graph showing the number of reads of various lengths. The numbers before and after are overlayed so that you can easily see how the trimming has affected the read lengths (right-click the graph to open it in a new view).

- **Trim settings** A summary of the settings used for trimming.
- **Detailed trim results**. A table with one row for each type of trimming:
  - **Input reads.** The number of reads used as input. Since the trimming is done sequentially, the number of retained reads from the first type of trim is also the number of input reads for the next type of trimming.
  - **No trim.** The number of reads that have been retained, unaffected by the trimming.
  - **Trimmed.** The number of reads that have been partly trimmed. This number plus the number from **No trim** is the total number of retained reads.
  - **Nothing left or discarded.** The number of reads that have been discarded either because the full read was trimmed off or because they did not pass the length trim (e.g. too short) or adapter trim (e.g. if **Discard when not found** was chosen for the adapter trimming).



1 Trim summary

Name	Number of reads	Avg.length	Number of reads after trim	Percentage trimmed	Avg.length after trim
reads	57.213	228,0	55.754	~100%	232,8

2 Read length before / after trimming

Figure 19.35: *A report with statistics on the trim results.*

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. This will start the trimming process.

If you trim paired data, the result will be a bit special. In the case where one part of a paired read has been trimmed off completely, you no longer have a valid paired read in your sequence list. In order to use paired information when doing assembly and mapping, the Workbench therefore creates two separate sequence lists: one for the pairs that are intact, and one for the single reads where one part of the pair has been deleted. When running assembly and mapping, simply select both of these sequence lists as input, and the Workbench will automatically recognize that one has paired reads and the other has single reads.

19.4 *De novo* assembly

The de novo assembly algorithm of *CLC Genomics Workbench* offers comprehensive support for a variety of data formats, including both short and long reads, and mixing of paired reads (both insert size and orientation).

The *de novo* assembly process has two stages:

1. First, simple contig sequences are created by using all the information that are in the read sequences. This is the actual *de novo* part of the process. These simple contig sequences do not contain any information about which reads the contigs are built from. This part is elaborated in section 19.4.1.

2. Second, all the reads are mapped using the simple contig sequence as reference. This is done in order to show e.g. coverage levels along the contigs and enabling more downstream analysis like SNP detection and creating mapping reports. Note that although a read aligns to a certain position on the contig, it does not mean that the information from this read was used for building the contig, because the mapping of the reads is a completely separate part of the algorithm.

If you wish to only have the simple contig sequences as output, this can be chosen when starting the de novo assembly (see section 19.4.5).

19.4.1 How it works

This section explains how the first stage of the de novo assembly works: CLC bio's de novo assembly algorithm works by using de Bruijn graphs. This is similar to how most new de novo assembly algorithms work. The basic idea is to make a table of all sub-sequences of a certain length (called words) found in the reads. The words are relatively short, e.g. about 20 for a bacterial genome and 27 for a human genome.

Given a word in the table, we can look up all the potential neighboring words (in all the examples here, word of length 16 are used) as shown in figure 19.36.

Backward neighbors	Starting word	Forward neighbors
AACGTAGCTAGCGCAT	ACGTAGCTAGCGCATG	CGTAGCTAGCGCATGA
CACGTAGCTAGCGCAT		CGTAGCTAGCGCATGC
GACGTAGCTAGCGCAT		CGTAGCTAGCGCATGG
TACGTAGCTAGCGCAT		CGTAGCTAGCGCATGT

Figure 19.36: *The word in the middle is 16 bases long, and it shares the 15 first bases with the backward neighboring word and the last 15 bases with the forward neighboring word.*

Typically, only one of the backward neighbors and one of the forward neighbors will be present in the table. A graph can then be made where each node is a word that is present in the table and edges connect nodes that are neighbors. This is called a de Bruijn graph.

For genomic regions without repeats or sequencing errors, we get long linear stretches of connected nodes. We may choose to reduce such stretches of nodes with only one backward and one forward neighbor into nodes representing sub-sequences longer than the initial words.

Figure 19.37 shows an example where one node has two forward neighbors:

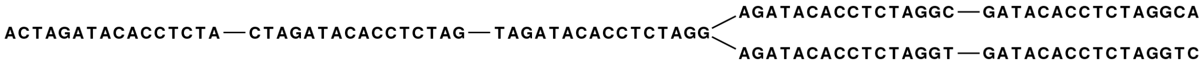


Figure 19.37: *Three nodes connected, each sharing 15 bases with its neighboring node and ending with two forward neighbors.*

After reduction, the three first nodes are merged, and the two sets of forward neighboring nodes are also merged as shown in figure 19.38.

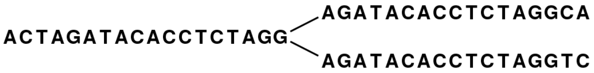


Figure 19.38: *The five nodes are compacted into three. Note that the first node is now 18 bases and the second nodes are each 17 bases.*

So bifurcations in the graph leads to separate nodes. In this case we get a total of three nodes after the reduction. Note that neighboring nodes still have an overlap (in this case 15 nucleotides since the word length is 16).

Given this way of representing the de Bruijn graph for the reads, we can consider some different situations:

When we have a *SNP or a sequencing error*, we get a so-called bubble as shown in figure 19.39.

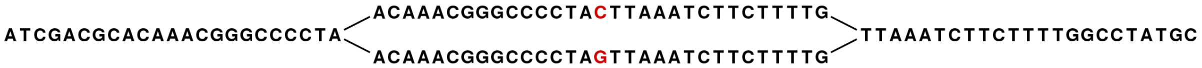

ATCGACGCACAAACGGGCCCCTA
ACAAACGGGCCCCTACTTAAATCTTCTTTTG
ACAAACGGGCCCCTAGTTAAATCTTCTTTTG
TTAAATCTTCTTTTGGCCTATGC

Figure 19.39: *A bubble caused by a SNP or a sequencing error.*

Here, the central position may be either a C or a G. If this was a sequencing error occurring only once, we would see that one path through the bubble will only be words seen a single time. On the other hand if this was a heterozygote SNP we would see both paths represented more or less equally. Thus, having information about how many times this particular word is seen in all the reads is very useful and this information is stored in the initial word table together with the words.

If we have a *repeat sequence* that is present twice in the genome, we would get a graph as shown in figure 19.40.

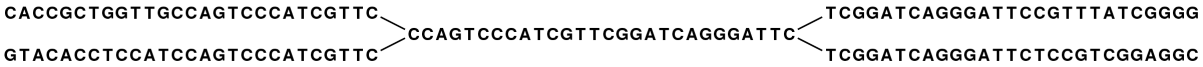

CACCGCTGGTTGCCAGTCCCATCGTTC
GTACACCTCCATCCAGTCCCATCGTTC
CCAGTCCCATCGTTCGGATCAGGGATTC
TCGGATCAGGGATTCCGTTTATCGGGG
TCGGATCAGGGATTCTCCGTCGGAGGC

Figure 19.40: *The central node represents the repeat region that is represented twice in the genome. The neighboring nodes represent the flanking regions of this repeat in the genome.*

Note that this repeat is 57 nucleotides long (the length of the sub-sequence in the central node above plus regions into the neighboring nodes where the sequences are identical). If the repeat had been shorter than 15 nucleotides, it would not have shown up as a repeat at all since the word length is 16. This is an argument for using long words in the word table. On the other hand, the longer the word, the more words from a read are affected by a sequencing error. Also, for each extra nucleotide in the words, we get one less word from each read. This is in particular an issue for very short reads. For example, if the read length is 35, we get 16 words out of each read of the word length is 20. If the word length is 25, we get only 11 words from each read.

To strike a balance, CLC bio's de novo assembler chooses a word length based on the amount of input data: the more data, the longer the word length.

A simple de novo assembly result would be to output the sequence of each reduced node. The bubbles described above from SNPs and sequencing errors as well as the repeats will make this quite a bad result with many short contigs. Instead, we can try to resolve the repeats with reads that span from a node before the repeat to a node after the repeat. Small bubbles can be resolved by choosing the path with the most coverage. Thus, by using the information from the full length reads, we are able to produce much longer contigs.

Furthermore, when paired reads are available, we can use this information to resolve even larger repeat regions that may not be spanned by individual reads, but are spanned by read pairs. This results in even longer contigs.

So in summary, the de novo assembly algorithm goes through these stages:

- Make a table of the words seen in the reads.
- Build de Bruijn graph from the word table.
- Use the reads to resolve the repeats.
- Use the information from paired reads to resolve larger repeats.
- Output resulting contigs based on the paths.

These stages are all performed by the assembler program.

Repeat regions in large genomes often get very complex: a repeat may be found thousands of times and part of one repeat may also be part of another repeat, further complicating the graph. Sometimes a repeat is longer than the read length (or the paired distance when pairs are available) and then it becomes impossible to resolve the repeat. This is simply because there is no information available about how to connect the nodes before the repeat to the nodes after the repeat. This means that no matter how much coverage we have, we will still get a number of separate contigs as a result.

19.4.2 Randomness in the results

A side-effect of the very compact data structures needed in order to keep the memory consumption low, is that the results will vary slightly from run to run, using the same data set. When counting the number of occurrences of a word, the assembler does not keep track of the exact number (which would consume a lot of memory) but uses an approximation which relies on some probability calculations. When using a multi-threaded CPU, the data structure is build in different ways for each run, and this means that the probability calculations for certain parts of the algorithm will be a bit different from run to run. This leads to differences in the results.

It should be noted that the differences are minor and will not affect the overall results. Keep in mind that whether you use CLC bio's assembler or other assemblers, there will never be one correct answer to the problem of de novo assembly. In this perspective, the small differences should not be considered a problem.

19.4.3 SOLiD data support in de novo assembly

SOLiD sequencing is done in color space. When viewed in nucleotide space this means that a single sequencing error changes the remainder of the read. An example read is shown in figure 19.41.

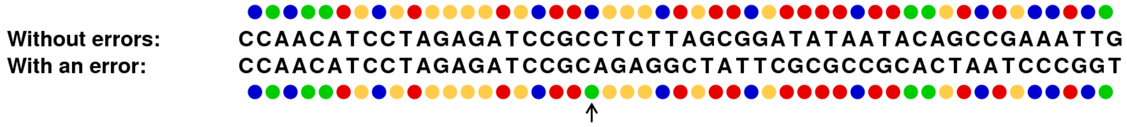

Without errors:
CCAACATCCTAGAGATCCGCCTCTTAGCGGATATAATACAGCCGAAATTG
With an error:
CCAACATCCTAGAGATCCGCAGAGGCTATTCGCGCCGCACTAATCCCGGT

Figure 19.41: *How an error in color space leads to a phase shift and subsequent problems for the rest of the read sequence*

Basically, this color error means that C's become A's and A's become C's. Likewise for G's and T's. For the three different types of errors, we get three different ends of the read. Along with the correct reads, we may get four different versions of the original genome due to errors. So if SOLiD reads are just regarded in nucleotide space, we get four different contig sequences with jumps from one to another every time there is a sequencing error.

Thus, to fully accommodate SOLiD sequencing data, the special nature of the technology has to be considered in every step of the assembly algorithm. Furthermore, SOLiD reads are fairly short and often quite error prone. Due to these issues, we have chosen not to include SOLiD support in the first algorithm steps, but only use the SOLiD data where they have a large positive effect on the assembly process: when applying paired information.

19.4.4 De novo assembly parameters

To start the assembly:

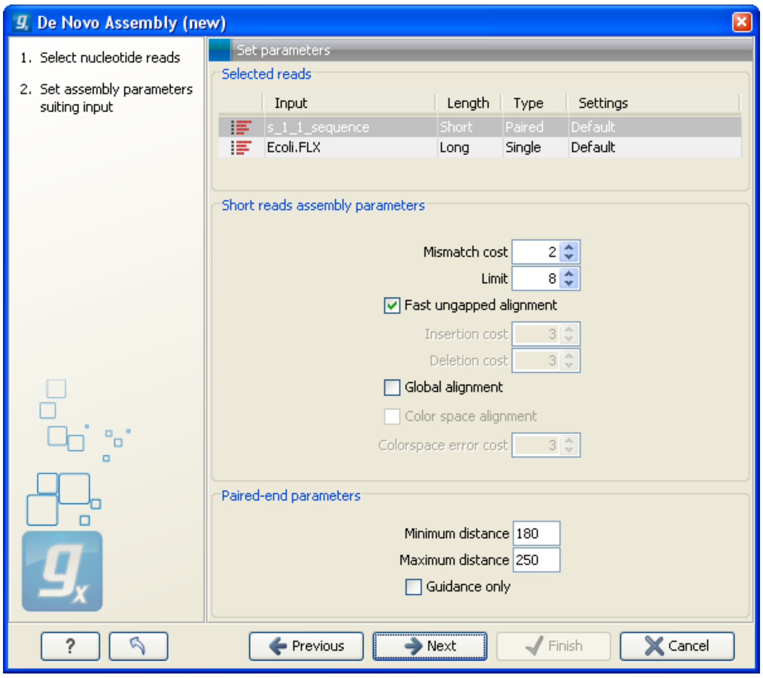
Toolbox | **High-throughput Sequencing (** **)** | **De Novo Assembly (** **)**

In this dialog, you can select one or more sequence lists or single sequences.

Click **Next** to set the parameters for the assembly. This will show a dialog similar to the one in figure 19.42.

Note that most of these parameters regard stage two of the de novo assembly where the reads are mapped back to the simple contig sequences. Please refer to section 19.5.3 for more information on the parameters and how to make selections of the data sets.

The only exception is the paired information in the panel at the bottom (this is only shown for paired data sets) which is used both for the first and second stage of the de novo assembly (see how this information is used in section 19.4.1). There is an additional option to use the data set for **Guidance only**. This is recommended for SOLiD data as explained in section 19.4.3. Note



1. Select nucleotide reads
2. Set assembly parameters suiting input

Selected reads

Input	Length	Type	Settings
s_1_1_sequence	Short	Paired	Default
Ecoli.FLX	Long	Single	Default

Short reads assembly parameters

Mismatch cost 2
Limit 8
☑ Fast ungapped alignment
Insertion cost 3
Deletion cost 3
□ Global alignment
□ Color space alignment
Colorspace error cost 3

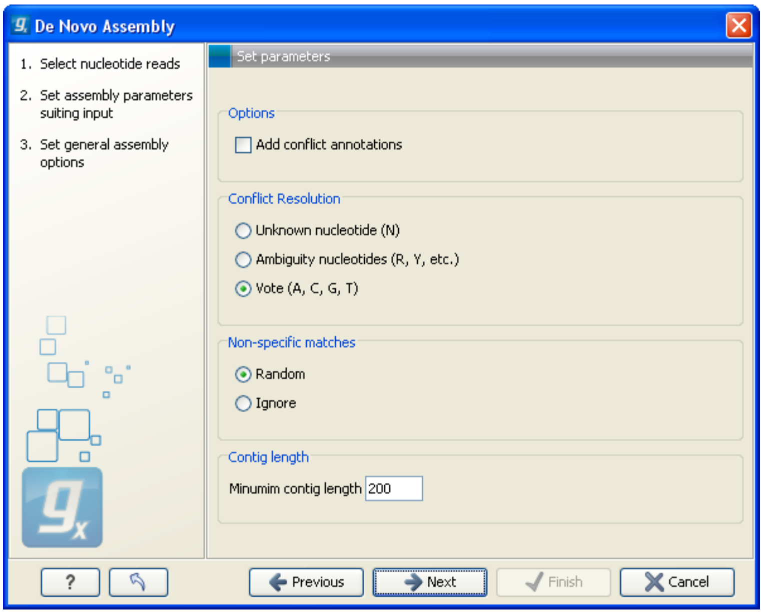
Paired-end parameters

Minimum distance 180
Maximum distance 250
□ Guidance only

Figure 19.42: *Setting parameters for the assembly.*

that if you need to have at least one data set where the guidance only option is not checked.

When you click **Next**, you will see the dialog shown in figure 19.43



1. Select nucleotide reads
2. Set assembly parameters suiting input
3. Set general assembly options

Options

□ Add conflict annotations

Conflict Resolution

○ Unknown nucleotide (N)
○ Ambiguity nucleotides (R, Y, etc.)
● Vote (A, C, G, T)

Non-specific matches

● Random
○ Ignore

Contig length

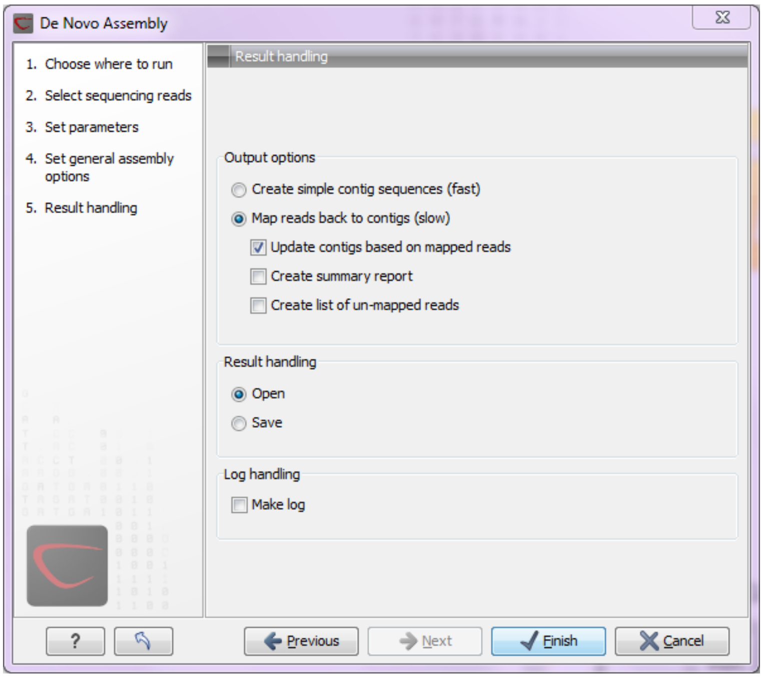
Minumim contig length 200

Figure 19.43: *Conflict resolution and annotation.*

All these parameters also part of the second read mapping stage and are explained in section 19.5.3. The only exception is that you can also specify the minimum contig length when doing de novo assembly. Contigs below this length will not be reported.

19.4.5 Assembly reporting options

Click **Next** lets you choose how the output of the assembly should be reported (see figure 19.44).



De Novo Assembly

1. Choose where to run
2. Select sequencing reads
3. Set parameters
4. Set general assembly options
5. Result handling

Result handling

Output options

○ Create simple contig sequences (fast)

● Map reads back to contigs (slow)

☑ Update contigs based on mapped reads

□ Create summary report

□ Create list of un-mapped reads

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 19.44: *Assembly reporting options.*

- **Create simple contig sequences.** This will show a sequence list containing all the consensus sequences from the assembly. **This option is much faster and less demanding for your computer.** This is explained in more detail in section 19.4.

- **Map reads back to contigs**. This will add mapping of the reads to the de novo assembly process. There is also an option to **Update contigs based on mapped reads**. This means that the original contig sequences produced from the de novo assembly will be updated to reflect the mapping of the reads (in most cases it will mean no change, but in some cases, the subsequent mapping step leads to new information). Note that if this option is selected, the contig lengths may get below the threshold specified in figure 19.43 because this threshold is applied to the original contig sequences. The result will be a table with one row per contig showing information about the contig length, and number of reads. Double-clicking one of the rows in the table will open the corresponding mapping. Furthermore, you can select a number of mappings and click the **Open Consensus/Contig** at the bottom of the table. That will open a sequence list of all the consensus/contig sequences.

  Furthermore, it will give some extra reporting options:

  - **Create Report**. This will generate a summary report as described in section 19.6.2.
  - **Create list of non-mapped reads**. This will put all the reads that could not be assembled into a sequence list.

Clicking **Finish** will start the assembly process. See section 18.6 for general information about viewing and editing the resulting mappings.

19.5 Map reads to reference

This section describes how to map a number of sequence reads to one or more reference sequences. When the reads come from a set of known sequences with relatively few variations, read mapping is often the right approach to assembling the data. The result of mapping reads to a reference is a "mapping" or a "mapping table" which is the term we use for an alignment of reads against a reference sequence.

19.5.1 Starting the read mapping

To start the read mapping:

Toolbox | **High-throughput Sequencing ( )** | **Map Reads to Reference ( )**

In this dialog, select the sequences or sequence lists containing the sequencing data. Note that the reference sequences should be selected in the next step.

When the sequences are selected, click **Next**, and you will see the dialog shown in figure 19.45.

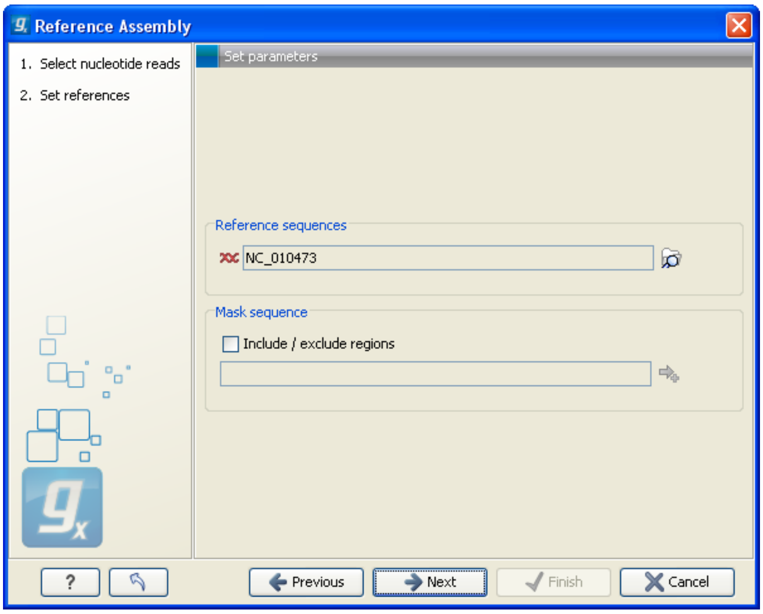


Figure 19.45: *Specifying the reference sequences and masking.*

At the top you select one or more reference sequences by clicking the **Browse and select element** ( ) button. You can select either single sequences or a list of sequences as reference sequences. When multiple reference sequence are used, the result of the mapping will be a mapping table with one entry per reference sequence.

19.5.2 Including or excluding regions (masking)

The next part of the dialog lets you *mask* the reference sequences. Masking refers to a mechanism where parts of the reference sequence are not considered in the mapping. This can be extremely useful for example when mapping human data, where more than 50 % of the sequence consists of repeats. Note that you should be careful masking all the repeat regions if your sequenced data contains the repeats. If you do that, some of the reads that would have matched a masked repeat region perfectly may be placed wrongly at another position even with a less-perfect match and lead to wrong results.

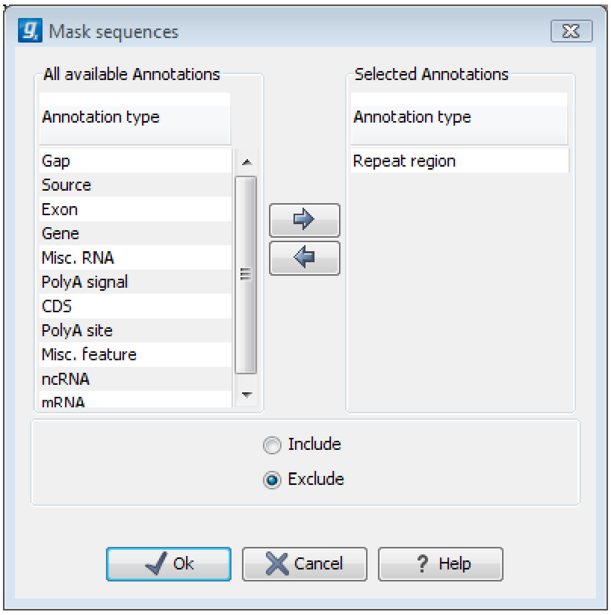
In order to mask e.g. repeat regions when doing read mapping, the repeat regions have to be annotated on the reference sequences.

Because the masking is based on annotations, any kind of annotations can be selected for masking. This means that you can choose to e.g. only map against the genes in the genome, or only the exons. As long as the reference sequences contain the relevant information in the form of annotations, it can be masked.

To mask a reference sequence, first click the **Include / exclude regions** checkbox, and second click the **Select annotation type** (⇨) button.

This will bring up a dialog with all the annotation types of the reference sequences listed to the left. Select one or more annotation types and click **Add** (⇨) button. Then select at the bottom whether you wish to **Include** or **Exclude** these annotation types. If you include, it means that only the regions covered by the selected type of annotations will be used in the read mapping. If you exclude, it means that all of the reference sequences except the regions covered by the selected type of annotations will be used in the read mapping.

You can see an example in figure 19.46.



Annotation type
Gap
Source
Exon
Gene
Misc. RNA
PolyA signal
CDS
PolyA site
Misc. feature
ncRNA
mRNA

Annotation type
Repeat region

○ Include

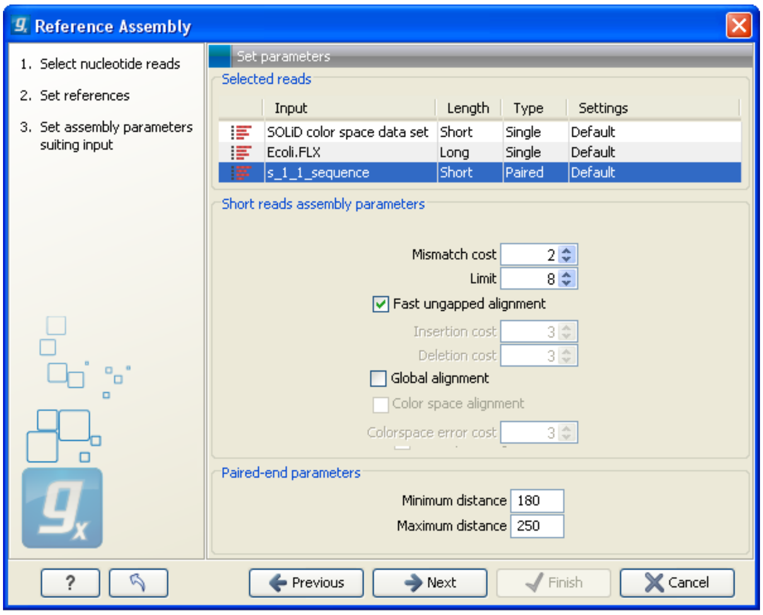
● Exclude

Figure 19.46: *Masking for repeats. The repeat region annotation type is selected and excluded in the mapping.*

19.5.3 Mapping parameters

Click **Next** to set the parameters for the mapping. This will show a dialog similar to the one in figure 19.47.

In order to understand what is going on here, a little explanation is needed: The *CLC Genomics Workbench* supports assembly of mixed data sets. This means that you can assemble and map both short reads, long reads, single reads and paired reads in one go. This makes it easy to combine the information from different sources, but it also makes the parameters a little more complex, because each data set may need its own parameters.



1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input

Selected reads

Input	Length	Type	Settings
SOLiD color space data set	Short	Single	Default
Ecoli.FLX	Long	Single	Default
s_1_1_sequence	Short	Paired	Default

Short reads assembly parameters

Mismatch cost 2

Limit 8

☑ Fast ungapped alignment

Insertion cost 3

Deletion cost 3

☐ Global alignment

☐ Color space alignment

Colorspace error cost 3

Paired-end parameters

Minimum distance 180

Maximum distance 250

Figure 19.47: *Setting parameters for mapping.*

At the top of the dialog shown in figure 19.47 is a table of all the sequence lists that were chosen in the first step. Clicking one of the lists shows the parameters that will be used this particular data set. Note that the Workbench automatically categorizes each of the lists into short/long reads and single/paired. Reads are considered short when they are less than 56 nucleotides, unless the data is in color space where the long reads algorithm is always applied regardless of the read length.

In the example in figure 19.47, you first adjust the parameters for the data set called `s_1_1_sequence` and then click the next data set called `Ecoli.FLX` and adjust parameters for this data set (as shown in figure 19.48). Because these data sets are different in terms of length and single/paired content, you have to set the parameters for each one. If you had two similar data sets, you could select both of them in the table and then change the settings for both.

Each of the parameters are described below:

Common parameters for short and long reads

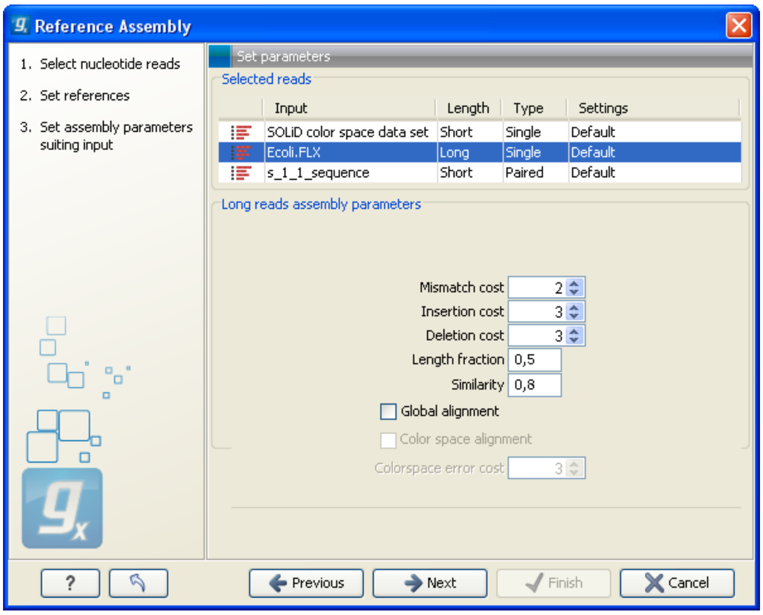
Three parameters are identical for both short and long reads:

Mismatch cost The cost of a mismatch between the read and the reference sequence.

Insertion cost The cost of an insertion in the read (causing a gap in the reference sequence)

Deletion cost The cost of having a gap in the read.

Global alignment Per default, the reads are aligned locally, allowing a number of "unaligned" nucleotides at the ends of the read. By selecting the global alignment option, you force the whole read to be aligned to the reference. Mismatches at the ends will then count as any other mismatch.



1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input

Set parameters

Selected reads

Input	Length	Type	Settings
SOLiD color space data set	Short	Single	Default
Ecoli.FLX	Long	Single	Default
s_1_1_sequence	Short	Paired	Default

Long reads assembly parameters

Mismatch cost 2
Insertion cost 3
Deletion cost 3
Length fraction 0,5
Similarity 0,8
□ Global alignment
□ Color space alignment
Colorspace error cost 3

Figure 19.48: *Setting parameters for the mapping.*

The score for a match is always 1.

Short reads parameters

For short reads, there is a threshold that determines whether the read should be included in the mapping:

Limit The relationship between the length of the read and the score. A limit of 8 (which is default) means that the total score for the alignment has to be more than the length of the read minus 8. This is explained in detail with examples below. This means that with the default costs, two mismatches, two deletions or two insertions will be allowed. If no mismatches or gaps are involved, it means that up to 8 unaligned nucleotides in the ends would be allowed. For very short reads, a limit of 5 could typically be used instead, allowing up to one mismatch and two unaligned nucleotides in the ends (or no mismatches and five unaligned nucleotides).

Given a certain quality threshold, it is possible to guarantee that all optimal ungapped alignments are found for each read. Alignments of short reads to reference sequences usually contain no gaps, so the short read assembly operates with a strict scoring threshold to allow the user to specify the amount of errors to accept.

With other short read mapping programs like Maq and Soap, the threshold is specified as the number of allowed mismatches. This works because those programs do global alignment. For local alignments it is a little more complicated.

The default alignment scoring scheme for short reads is +1 for matches and -2 for mismatches. The limit for accepting an alignment is given as the alignment score relative to the read length. For example, if the score limit is 8 below the length, up to two mismatches are allowed as well as two ending nucleotides not assembled (remember that a mismatch costs 2 points, but when there is a mismatch, a potential match is also lost). Alternatively, with one mismatch, up to 5 unaligned positions are allowed. Or finally, with no mismatches, up to 8 unaligned positions are allowed. See figure 19.49 for examples. The default setting is exactly this limit of 8 below the length.

```
CGTATCAATCGATTACGCTATGAATG           CGTATCAATCGATTACGCTATGAATG
   ||||||||||||||||||||      20          |||||||||||||||||||      19
   ATCAATCGATTACGCTATGA                 TTCAATCGATTACGCTATGA

CGTATCAATCGATTACGCTATGAATG           CGTATCAATCGATTACGCTATGAATG
   |||||||| |||||||||||      17          ||||||| |||||||||||      16
   ATCAATCGGTTACGCTATGA                 TTCAATCGGTTACGCTATGA

CGTATCAATCGATTACGCTATGAATG           CGTATCAATCGATTACGCTATGAATG
    ||||||| ||||||||||       15         ||||| || |||||||||||      14
   CTCAATCGGTTACGCTATGA                 ATCAACCGGTTACGCTATGA

CGTATCAATCGATTACGCTATGAATG           CGTATCAATCGATTACGCTATGAATG
    ||||||| |||| ||||||      13          |||||||||| ||||          12
   TTCAATCGGTTACCCTATGA                 ATCAATCGATTGCGCTCTTT

CGTATCAATCGATTACGCTATGAATG           CGTATCAATCGATTACGCTATGAATG
    ||||||| |||| |||||       12             ||||||||||||          12
   TTCAATCGGTTACCCTATGC                 AGCTATCGATTACGCTCTTT
```

Figure 19.49: Examples of ungapped alignments allowed for a 20 bp read with a scoring limit of 8 below the length using the default scoring scheme. The scores are noted to the right of each alignment. For reads this short, a limit of 5 would typically be used instead, allowing up to one mismatch and two unaligned nucleotides in the ends (or no mismatches and five unaligned nucleotides).

Note that if you choose to do global alignment, the default setting means that up to two mismatches are allowed (because "unaligned positions" at the ends are counted as mismatches as well).

The match score is always +1. If the mismatch cost is changed, the default score limit will also change to:

$$score\ limit = 3 \times (1 + mismatch\ cost) - 1$$

The default mismatch *score* of -2 equals a mismatch *cost* of 2 and a score limit of 8 below the read length, as stated above. For any mismatch cost, the default score limit allows any alignment scoring strictly better than 3 mismatches.

The maximum score limit also depends on the mismatch cost:

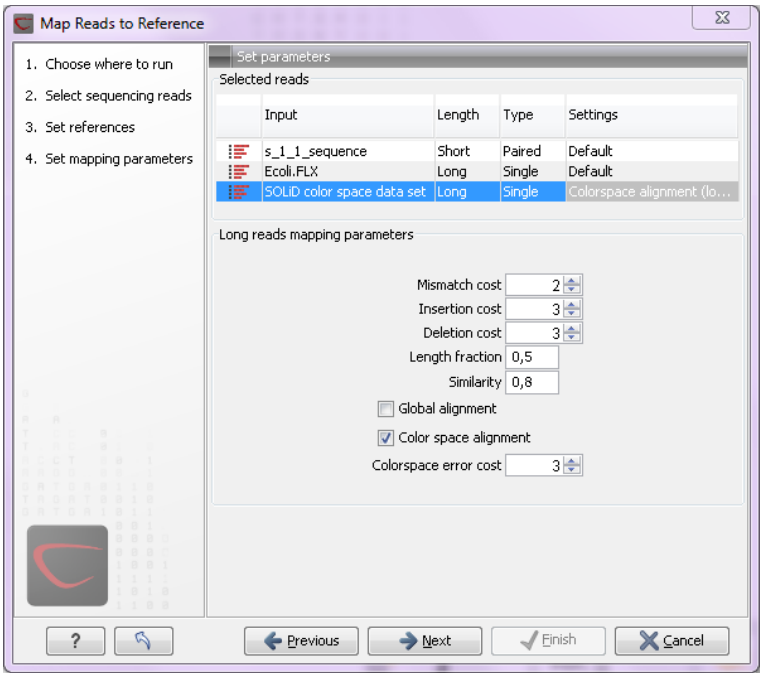
$$max\ score\ limit = 4 \times (1 + mismatch\ cost) - 1$$

Gapped alignment is also allowed for short reads. Contrary to ungapped alignments, it is very difficult to guarantee that all gapped alignments of a certain quality are found. The scoring limit discussed above applies to both gapped and ungapped alignments and there is a guarantee that there are no ungapped exceeding the limit, but there is is no such guarantee for gapped alignments. This being said, the program does a good effort to find the best gapped alignments and usually succeeds. Besides the limit, there are also two options related to mapping of color space data (from SOLiD systems). If you do not have color space data, these will be disabled and are not relevant.

Color space alignment This will determine if mapping is to be performed in color space. This is strongly recommended for SOLiD data.

Color error cost The cost of a color error.

An example of a color space data set is shown in figure 19.50.



1. Choose where to run
2. Select sequencing reads
3. Set references
4. Set mapping parameters

Selected reads

Input	Length	Type	Settings
s_1_1_sequence	Short	Paired	Default
Ecoli.FLX	Long	Single	Default
SOLiD color space data set	Long	Single	Colorspace alignment (lo...

Long reads mapping parameters

Mismatch cost 2
Insertion cost 3
Deletion cost 3
Length fraction 0,5
Similarity 0,8
☐ Global alignment
☑ Color space alignment
Colorspace error cost 3

Figure 19.50: *Setting parameters for the mapping.*

For more details about this, please see section 19.8 which explains how color space mapping is performed in greater detail.

Long reads parameters

For long reads, the read mapping as two stages: First, the optimal alignment of the read is found, based on the costs specified above (e.g. to favor mismatches over indels). Second, a filtering process determines whether this match is good enough for the read to be included in the alignment. The filtering threshold is determined by two fractions:

Length fraction Set minimum length fraction of a read that must match the reference sequence. Setting a value at 0.5 means that at least half the read needs to match the reference sequence for the read to be included in the final mapping.

Similarity Set minimum fraction of identity between the read and the reference sequence. If you want the reads to have e.g. at least 90% identity with the reference sequence in order to be included in the final mapping, set this value to 0.9. Note that the similarity fraction does not apply to the whole read; it relates to the Length fraction. With the default values, it means that *at least 50 % of the read must have at least 90 % identity*.

Paired reads

At the bottom you can specify how **Paired reads** should be handled. You can read more about how paired data is imported and handled in section 19.1. If the sequence list used as input for the assembly contains paired reads, this option will automatically be shown - if it contains single reads, this option will not be shown.

For the paired reads, you can specify a distance interval between the two sequences in a pair. This will be used for the assembly process to determine how far it can expect the two reads to be from each other. This value should include the length of the sequences as well (not just the distance in between). If you set this value very precisely, you will miss some of the opportunities to detect genomic rearrangements as described in section 19.9.3. On the other hand, a precise distance interval will give a more accurate assembly in the places where there are not significant variation between the sequencing data and the reference sequence.

The approach taken for determining the placement of read pairs is the following:

- First, all the optimal placements for the two individual reads are found.
- Then, the allowed placements according to the paired distance interval are found.
- If both reads can be placed independently but no pairs satisfies the paired criteria, the reads are treated as independent and not marked as a pair.
- If only one pair of placements satisfy the criteria, the reads are placed accordingly and marked as uniquely placed even if either read may have multiple optimal placements.
- If several placements satisfy the paired criteria, the read is treated as a "non-specific match" (see section 19.5.4 for more information.)

By default, mapping is done with local alignment of reads to a set of reference sequences. The advantage of performing local alignment instead of global alignment is that the ends are automatically removed if there are sufficiently many sequencing errors there. If the ends of the reads contain vector contamination or adapter sequences, local alignment is also desirable. Note that the aligned region has to be greater than the length threshold set.

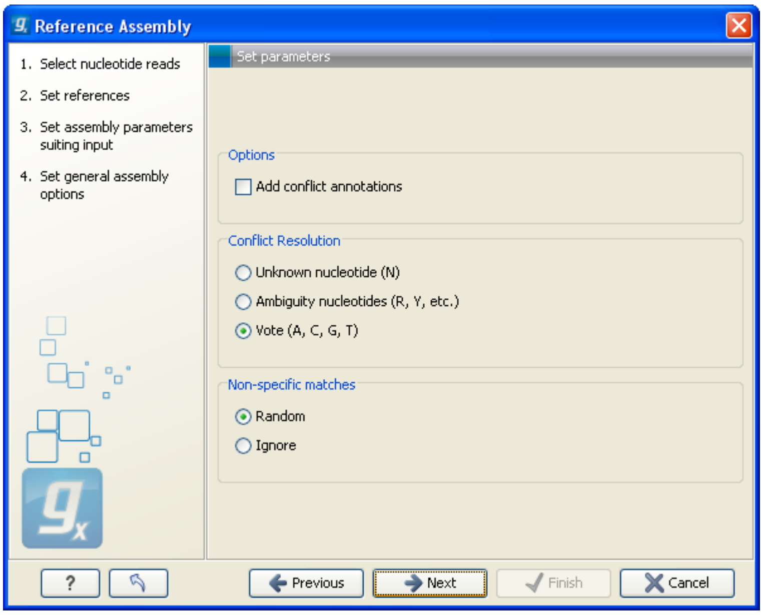
19.5.4 General mapping options

When you click **Next**, you will see the dialog shown in figure 19.51

At the top, you can choose to **Add conflict annotations** to the consensus sequence. Note that there may be a huge number of annotations and that it may give a visually cluttered overview of the mapping of the reads. A subtle detail about the annotations - if you add annotations, you will be able to see *resolved* conflicts in the table view (▦) of the mapping (e.g. if you edit the bases after mapping). If there are no annotations, only the non-resolved conflicts are shown.

If there is a conflict between reads, i.e. a position where there is disagreement about which base is correct, you can specify how the consensus sequence should reflect the conflict:

- **Vote (A, C, G, T).** The conflict will be solved by counting instances of each nucleotide and then letting the majority decide the nucleotide in the consensus. In case of equality, ACGT are given priority over one another in the stated order.
- **Unknown nucleotide (N).** The consensus will be assigned an 'N' character in all positions with conflicts.


Reference Assembly
1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input
4. Set general assembly options
Set parameters
?
Previous
Next
Finish
Cancel

Options
□ Add conflict annotations

Conflict Resolution
○ Unknown nucleotide (N)
○ Ambiguity nucleotides (R, Y, etc.)
● Vote (A, C, G, T)

Non-specific matches
● Random
○ Ignore

Figure 19.51: *Conflict resolution and annotation.*

- **Ambiguity nucleotides (R, Y, etc.).** The consensus will display an ambiguity nucleotide reflecting the different nucleotides found in the reads. For an overview of ambiguity codes, see Appendix I.

At the bottom of the dialog, you can specify how **Non-specific matches** should be treated. The concept of Non-specific matches refers to a situation where a read aligns at more than one position. In this case you have two options:

- **Random**. This will place the read in one of the positions randomly.
- **Ignore**. This will not include the read in the final mapping.

19.5.5 Assembly reporting options

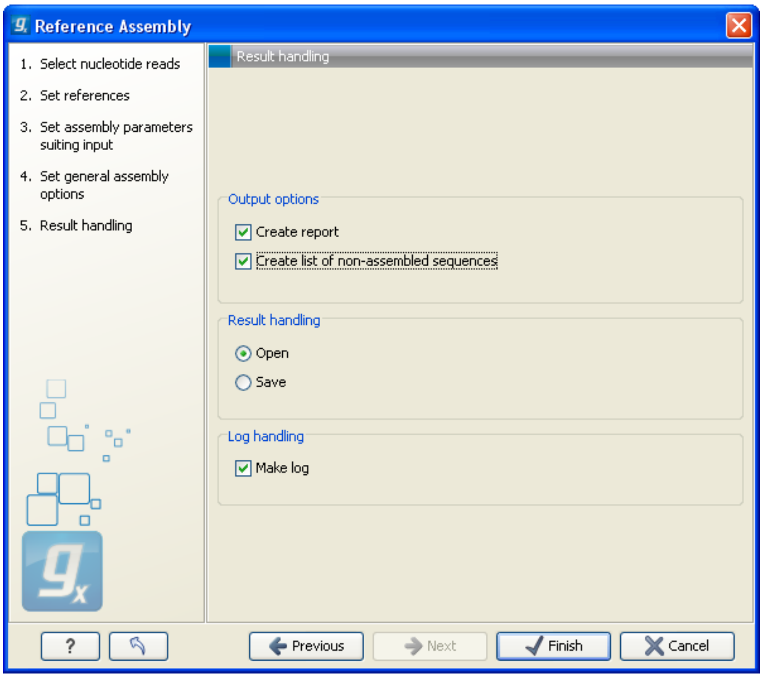
Click **Next** lets you choose how the output of the assembly should be reported (see figure 19.52).

First, you can choose to save or open the results, and if you wish to see a log of the process (see section 9.2).

No matter what you choose, you will always see the visual read mapping, but in addition you have two extra output options:

- **Create Report**. This will generate a summary report as described in section 19.6.2.
- **Create list of non-mapped sequences**. This will put all the reads that could not be mapped to the reference into a sequence list.

If you have used more than one reference sequence, the Workbench creates a table which makes it easier to get an overview. The table includes this information:



1. Select nucleotide reads
2. Set references
3. Set assembly parameters suiting input
4. Set general assembly options
5. Result handling

Result handling

Output options

☑ Create report

☑ Create list of non-assembled sequences

Result handling

● Open

○ Save

Log handling

☑ Make log

Figure 19.52: *Assembly reporting options.*

1. Length of reference sequence
2. Length of consensus sequence
3. Number of reads
4. Average coverage
5. Total number of conflicts

Double-clicking one of the rows in the table will open the corresponding mapping. Furthermore, you can select a number of rows and click the **Open Consensus** at the bottom of the table. That will open a sequence list of all the consensus sequences of the selected rows.

Clicking **Finish** will start the mapping. See section 18.6 for general information about viewing and editing the resulting mappings. For special information about genome-size mapping, see section 19.9.

19.6 Mapping reports

You can create two kinds of reports regarding read mappings and de novo assemblies: *First*, you can choose to generate a summary report about the mapping process itself (see sections 19.4.5 and 19.5.5). This report is described in section 19.6.2 below. *Second*, you can generate a detailed statistics report after the mapping or assembly has finished. This report is useful if you want to generate statistics across results made in different processes, and it generates more detailed statistics than the summary mapping report. This report is described below.

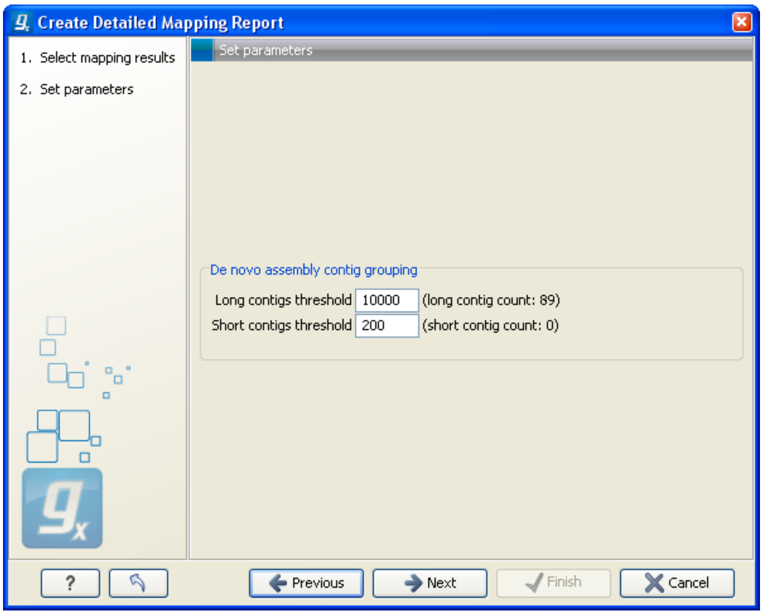
19.6.1 Detailed mapping report

To create a detailed mapping report:

Toolbox | High-throughput Sequencing ( ) | Create Detailed Mapping Report ( )

This opens a dialog where you can select mapping results ( )/ ( ) or RNA-Seq analysis results ( ) (see sections 19.4 and 19.5 for information on how to create a contig and section 19.14 for information on how to create RNA-Seq analysis results).

Clicking **Next** will display the dialog shown in figure 19.53



De novo assembly contig grouping

Long contigs threshold 10000 (long contig count: 89)

Short contigs threshold 200 (short contig count: 0)

Figure 19.53: *Parameters for mapping reports.*

The first option is to set thresholds for grouping long and short contigs. The grouping is used to show statistics like number of contigs, mean length etc for the contigs in each group. This is only relevant for de novo assemblies. Note that the de novo assembly in the *CLC Genomics Workbench* per default only reports contigs longer than 200 bp (this can be changed when running the assembly).

Click **Next** to select output options as shown in figure 19.54

Per default, an overall report will be created as described below. In addition, by checking **Create table with statistics for each reference** you can create a table showing detailed statistics for each reference sequence (for de novo results the contigs act as reference sequences, so it will be one row per contig). The following sections describe the information produced.

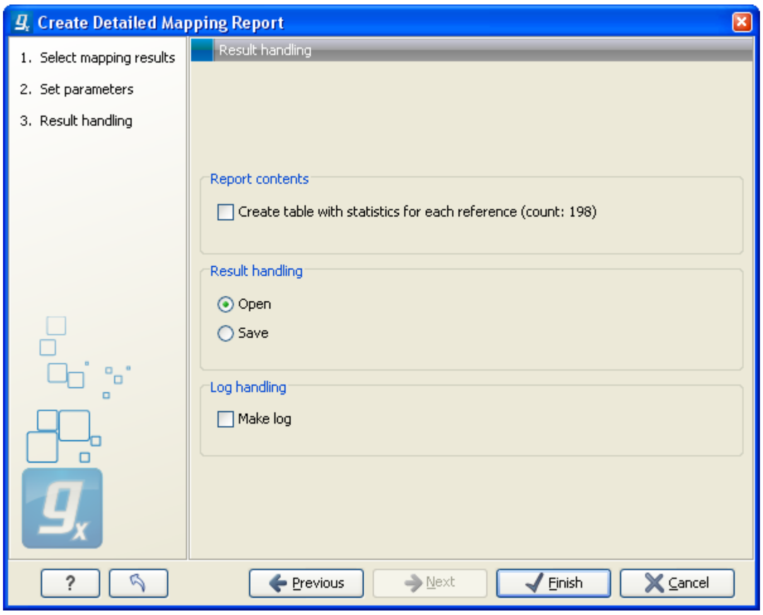
Reference sequence statistics

For reports on results of read mapping, section two concerns the reference sequences. The reference identity part includes the following information:

Reference name The name of the reference sequence.

Reference Latin name The reference sequence's Latin name.

Reference description Description of the reference.



Create Detailed Mapping Report

1. Select mapping results
2. Set parameters
3. Result handling

Result handling

Report contents

□ Create table with statistics for each reference (count: 198)

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 19.54: *Optionally create a table with detailed statistics per reference.*

If you want to inspect and edit this information, right-click the reference sequence in the contig and choose **Open This Sequence** and switch to the **Element info** (![]) tab (learn more in section 10.4). Note that you need to create a new report if you want the information in the report to be updated. If you update the information for the reference sequence within the contig, you should know that it doesn't affect the original reference sequence saved in the **Navigation Area**.

The next part of the report reports coverage statistics including GC content of the reference sequence. Note that coverage is reported on two levels: including and excluding zero coverage regions. In some cases, you do not expect the whole reference to be covered, and only the coverage levels of the covered parts of the reference sequence are interesting. On the other hand, if you have sequenced the full genome that you use as reference, the overall coverage is probably the most relevant number (i.e. including zero coverage regions).

A position on the reference is counted as "covered" when at least one read is aligned to it. Note that unaligned ends (faded nucleotides at the ends) that are produced when mapping using local alignment do not contribute to the coverage. In the example shown in figure 19.55, there is a region of zero coverage in the middle and one time coverage on each side. Note that the gaps to the very right are within the same read which means that these two positions on the reference sequence are still counted as "covered".

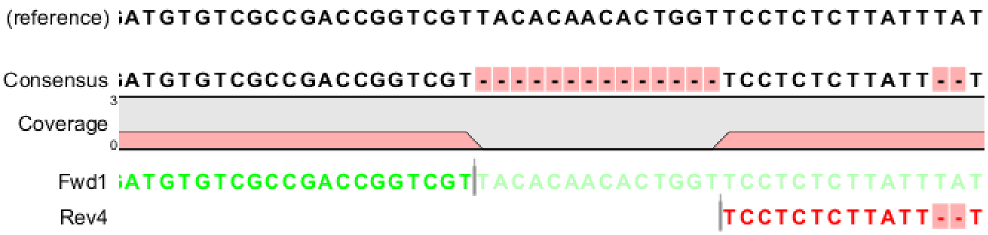

(reference) ATGTGTCGCCGACCGGTCGTTACACAACACTGGTTCCTCTCTTATTTAT
Consensus ATGTGTCGCCGACCGGTCGT--------------TCCTCTCTTATT--T
3
Coverage
0
Fwd1 ATGTGTCGCCGACCGGTCGTTACACAACACTGGTTCCTCTCTTATTTAT
Rev4 TCCTCTCTTATT--T

Figure 19.55: *A region of zero coverage in the middle and one time coverage on each side. Note that the gaps to the very right are within the same read which means that these two positions on the reference sequence are still counted as "covered".*

The identity section is followed by some statistics on the zero-coverage regions; the number, minimum and maximum length, mean length, standard deviation, total length and a list of the regions. If there are too many regions, they will not all be listed in the report (if there are more than 20, only the first 10 are reported).

Next follow two bar plots showing the distribution of coverage with coverage level on the x-axis and number of contig positions with that coverage on the y-axis. An example is shown in figure 19.56.

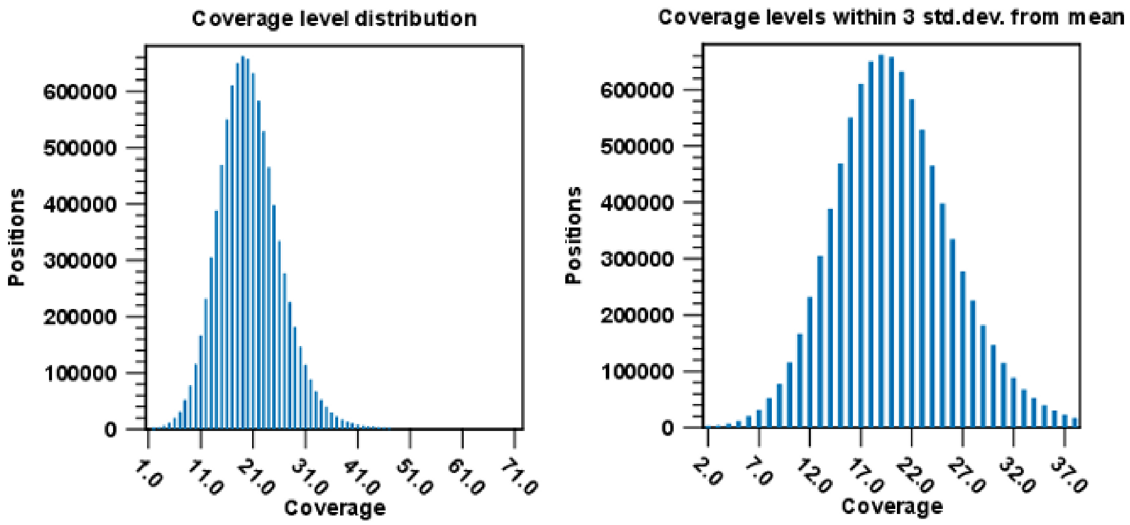


Figure 19.56: *Distribution of coverage - to the left for all the coverage levels, and to the right for coverage levels within 3 standard deviations from the mean.*

The graph to the left shows all the coverage levels, whereas the graph to the right shows coverage levels within 3 standard deviations from the mean. The reason for this is that for complex genomes, you will often have a few regions with extremely high coverage which will affect the resolution of the graph, making it impossible to see the coverage distribution for the majority of the contigs. These coverage outliers are excluded when only showing coverage within 3 standard deviations from the mean. Note that zero-coverage regions are not shown in the graph but reported in text below (this information is also in the zero-coverage section). Below the second coverage graph there are some statistics on the data that is outside the 3 standard deviations.

One of the biases seen in sequencing data concerns GC content. Often there is a correlation between GC content and coverage. In order to investigate this correlation, the report includes a graph plotting coverage against GC content (see figure 19.57). Note that you can see the GC content for each reference sequence in the table above.

The plot displays, for each GC content level (0-100 %), the mean read coverage of 100bp reference segments with that GC content.

At the end follows statistics about the reads which are the same for both reference and de novo assembly (see section 19.6.1 below).

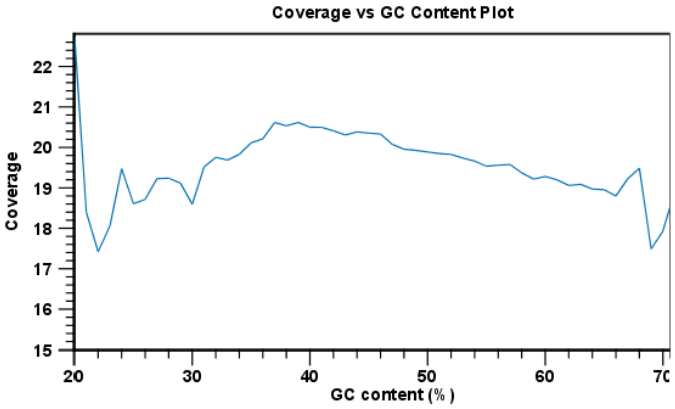


Figure 19.57: *The plot displays, for each GC content level (0-100 %), the mean read coverage of 100bp reference segments with that GC content.*

Contig statistics for de novo assembly

After the summary there is a section about the contig lengths. For each set of contigs, you can see the number of contigs, minimum, maximum and mean lengths, standard deviation and total contig length (sum of the lengths of all contigs in the set). The contig sets are:

N25 contigs The N25 contig set is calculated by summarizing the lengths of the biggest contigs until you reach 25 % of the total contig length. The minimum contig length in this set is the number that is usually used to report the N25 value of a de novo assembly.

N50 This measure is similar to N25 - just with 50 % instead of 25 %. This is probably the most well-known measure of de novo assembly quality - it is a more informative way of measuring the lengths of contigs.

N75 Similar to the ones above, just with 75 %.

All contigs All contigs that were selected.

Long contigs This contig set is based on the threshold set in the dialog in figure 19.53.

Short contigs This contig set is based on the threshold set in the dialog in figure 19.53. Note that the de novo assembly in the *CLC Genomics Workbench* per default only reports contigs longer than 200 bp.

Next follow two bar plots showing the distribution of coverage with coverage level on the x-axis and number of contig positions with that coverage on the y-axis. An example is shown in figure 19.58.

The graph to the left shows all the coverage levels, whereas the graph to the right shows coverage levels within 3 standard deviations from the mean. The reason for this is that for complex genomes, you will often have a few regions with extremely high coverage which will affect the resolution of the graph, making it impossible to see the coverage distribution for the majority of the contigs. These coverage outliers are excluded when only showing coverage within 3 standard deviations from the mean. Below the second coverage graph there are some statistics on the data that is outside the 3 standard deviations. At the end follows statistics about the reads which are the same for both reference and de novo assembly (see section 19.6.1 below).

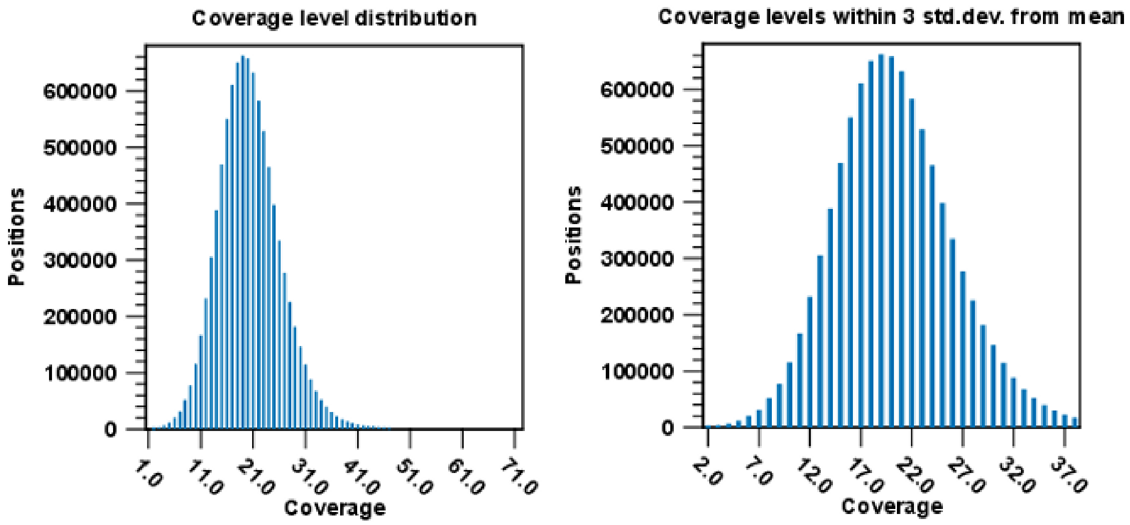


Figure 19.58: *Distribution of coverage - to the left for all the coverage levels, and to the right for coverage levels within 3 standard deviations from the mean.*

Read statistics

This section contains simple statistics for all mapped reads, non-specific matches (reads that match more than place during the assembly), non-perfect matches and paired reads. **Note!** Paired reads are counted as two, even though they form one pair. The section on paired reads also includes information about paired distance and counts the number of pairs that were broken due to:

Wrong distance When starting the mapping, a distance interval is specified. If the reads during the mapping are placed outside this interval, they will be counted here.

Mate inverted If one of the reads has been matched as reverse complement, the pair will be broken (note that the pairwise orientation of the reads is determined during import).

Mate on other contig If the reads are placed on different contigs, the pair will also be broken.

Mate not matched If only one of the reads match, the pair will be broken as well.

Below these tables follow two graphs showing distribution of paired distances (see figure 19.59) and distribution of read lengths. Note that the distance includes both the read sequence and the insert between them.

19.6.2 Summary assembly and mapping report

If you choose to create a report as part of the read mapping or assembly (see sections 19.4.5 and 19.5.5), this report will summarize the results of the assembly process. An example of a report is shown in figure 19.60

The information included in the report is:

- **Summary statistics**. A summary of the mapping statistics:

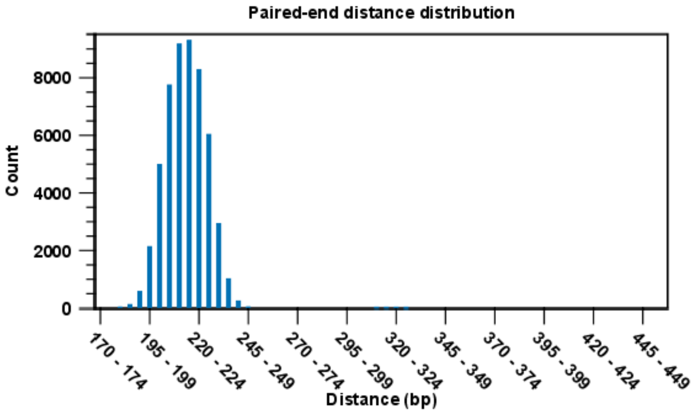
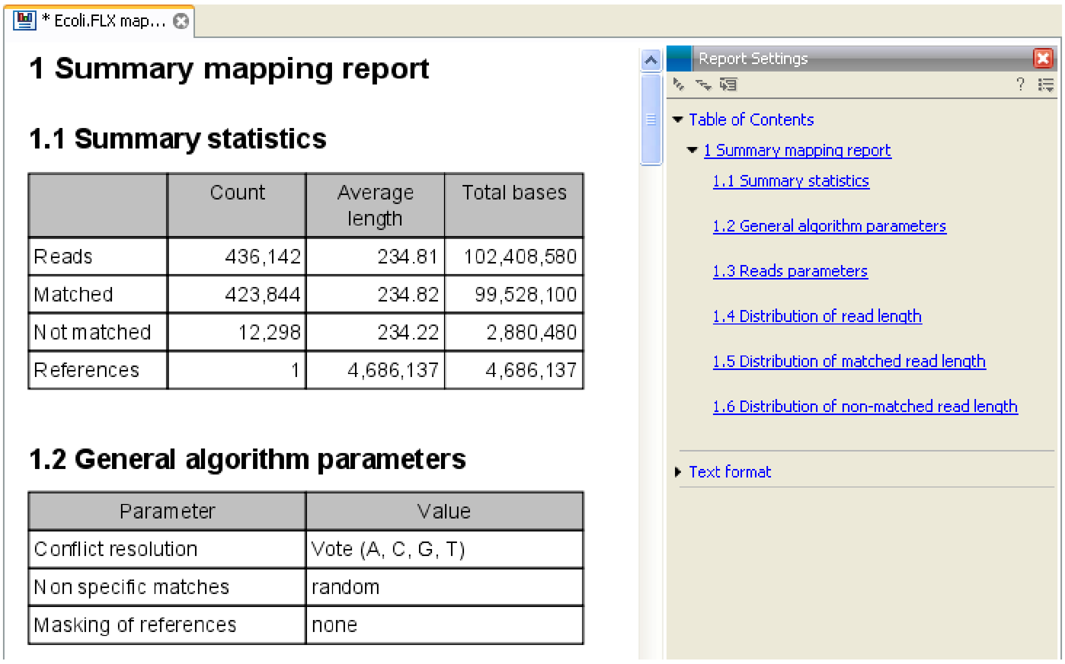


Figure 19.59: *A bar plot showing the distribution of distances between intact pairs.*


* Ecoli.FLX map...
Report Settings
Table of Contents
1 Summary mapping report
1.1 Summary statistics
1.2 General algorithm parameters
1.3 Reads parameters
1.4 Distribution of read length
1.5 Distribution of matched read length
1.6 Distribution of non-matched read length
Text format

1 Summary mapping report

1.1 Summary statistics

	Count	Average length	Total bases
Reads	436,142	234.81	102,408,580
Matched	423,844	234.82	99,528,100
Not matched	12,298	234.22	2,880,480
References	1	4,686,137	4,686,137

1.2 General algorithm parameters

Parameter	Value
Conflict resolution	Vote (A, C, G, T)
Non specific matches	random
Masking of references	none

Figure 19.60: *The summary mapping report.*

- **Reads**. The number of reads and the average length.
- **Mapped**. The number of reads that are mapped and their average length.
- **Not mapped**. The number of reads that do not map and their average length.
- **References/contigs**. Number of reference sequences or contigs (for de novo assembly)

- **Parameters**. The settings used are reported for the process as a whole and for each sequence list used as input.
- **Quality**. For de novo assemblies, the N25, N50 and N75 numbers are reported (learn more about this in section 19.6.1).
- **Distribution of read length**. For each sequence length, you can see the number of reads and the distribution in percent. This is mainly useful if you don't have too much variance in the lengths as you have in e.g. Sanger sequencing data.

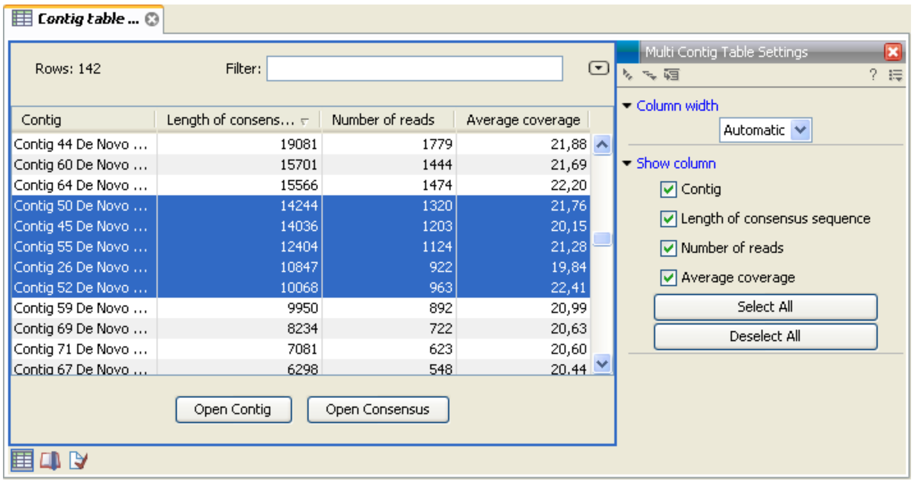
- **Distribution of matched reads lengths**. Equivalent to the above, except that this includes only the reads that have been matched to a contig.
- **Distribution of non-matched reads lengths**. Show the distribution of lengths of the rest of the sequences.

You can copy the information from the report by selecting in the report and click **Copy** ( ). You can also export the report in Excel format.

19.7 Mapping table

When several reference sequence are used or you are performing de novo assembly with the reads mapped back to the contig sequences, (see sections 19.4.5 and 19.5.5), all your mapping data will be accessible from a table ( ). It means that all the individual mappings are treated as one single file to be saved in the **Navigation Area** as a table.

An example of a mapping table for a *de novo* assembly is shown in figure 19.61.



Rows: 142 Filter:

Contig	Length of consens...	Number of reads	Average coverage
Contig 44 De Novo ...	19081	1779	21,88
Contig 60 De Novo ...	15701	1444	21,69
Contig 64 De Novo ...	15566	1474	22,20
Contig 50 De Novo ...	14244	1320	21,76
Contig 45 De Novo ...	14036	1203	20,15
Contig 55 De Novo ...	12404	1124	21,28
Contig 26 De Novo ...	10847	922	19,84
Contig 52 De Novo ...	10068	963	22,41
Contig 59 De Novo ...	9950	892	20,99
Contig 69 De Novo ...	8234	722	20,63
Contig 71 De Novo ...	7081	623	20,60
Contig 67 De Novo ...	6298	548	20,44

Column width: Automatic

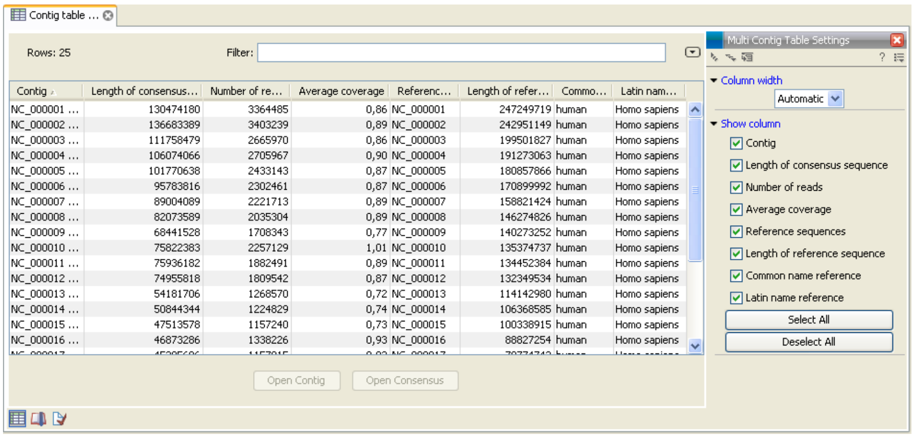
Show column
☑ Contig
☑ Length of consensus sequence
☑ Number of reads
☑ Average coverage

Figure 19.61: *The mapping table.*

The information included in the table is:

- **Name**. When mapping reads to a reference, this will be the name of the reference sequence.
- **Length of consensus sequence.** The length of the consensus sequence. Subtracting this from the length of the reference will indicate how much of the reference that has not been covered by reads.
- **Number of reads**. The number of reads. Reads hitting multiple places on different reference sequences are placed according to your input for **Non-specific matches**
- **Average coverage**. This is simply summing up the bases of the aligned part of all the reads divided by the length of the reference sequence.

For read mapping, there is more information taken from the reference sequence used as input. An example of a contig table produced by mapping reads to a reference is shown in figure 19.62.



Contig table ...

Rows: 25 Filter:

Contig	Length of consensus...	Number of re...	Average coverage	Referenc...	Length of refer...	Commo...	Latin nam...
NC_000001 ...	130474180	3364485	0,86	NC_000001	247249719	human	Homo sapiens
NC_000002 ...	136683389	3403239	0,89	NC_000002	242951149	human	Homo sapiens
NC_000003 ...	111758479	2665970	0,86	NC_000003	199501827	human	Homo sapiens
NC_000004 ...	106074066	2705967	0,90	NC_000004	191273063	human	Homo sapiens
NC_000005 ...	101770638	2433143	0,87	NC_000005	180857866	human	Homo sapiens
NC_000006 ...	95783816	2302461	0,87	NC_000006	170899992	human	Homo sapiens
NC_000007 ...	89004089	2221713	0,89	NC_000007	158821424	human	Homo sapiens
NC_000008 ...	82073589	2035304	0,89	NC_000008	146274826	human	Homo sapiens
NC_000009 ...	68441528	1708343	0,77	NC_000009	140273252	human	Homo sapiens
NC_000010 ...	75822383	2257129	1,01	NC_000010	135374737	human	Homo sapiens
NC_000011 ...	75936182	1882491	0,89	NC_000011	134452384	human	Homo sapiens
NC_000012 ...	74955818	1809542	0,87	NC_000012	132349534	human	Homo sapiens
NC_000013 ...	54181706	1268570	0,72	NC_000013	114142980	human	Homo sapiens
NC_000014 ...	50844344	1224829	0,74	NC_000014	106368585	human	Homo sapiens
NC_000015 ...	47513578	1157240	0,73	NC_000015	100338915	human	Homo sapiens
NC_000016 ...	46873286	1338226	0,93	NC_000016	88827254	human	Homo sapiens

Open Contig Open Consensus

Multi Contig Table Settings

Column width: Automatic

Show column
☑ Contig
☑ Length of consensus sequence
☑ Number of reads
☑ Average coverage
☑ Reference sequences
☑ Length of reference sequence
☑ Common name reference
☑ Latin name reference

Select All
Deselect All

Figure 19.62: *The contig table.*

Besides the information that is also in the *de novo* table, there is information about name, common name and Latin name of each reference sequence.

At the bottom of the table, there are two buttons which apply to the rows that you select (press Ctrl + A - ⌘ +A on Mac - to select all):

- **Open Mapping**. Simply opens the read mapping for visual inspection. You can also open one mapping simply by double-clicking in the table.
- **Open Consensus**. Creates a sequence list of all the consensus sequences. This can be used for further analysis or exported (📂) in e.g. fasta format.

You can copy the textual information from the table by selecting in the table and click **Copy** (📋). This can then be pasted into e.g. Excel. You can also export the table in Excel format.

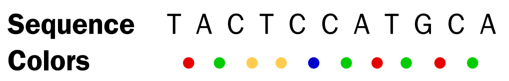
19.8 Color space

19.8.1 Sequencing

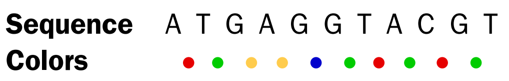
The SOLiD sequencing technology from Applied Biosystems is different from other sequencing technologies since it does not sequence one base at a time. Instead, two bases are sequenced at a time in an overlapping pattern. There are 16 different dinucleotides, but in the SOLiD technology, the dinucleotides are grouped in four carefully chosen sets, each containing four dinucleotides. The colors are as follows:

Base 1	**Base 2**			
	A	C	G	T
A	🔵	🟢	🟡	🔴
C	🟢	🔵	🔴	🟡
G	🟡	🔴	🔵	🟢
T	🔴	🟡	🟢	🔵

Notice how a base and a color uniquely defines the following base. This approach can be used to deduce a whole sequence from the initial nucleotide and a series of colors. Here is a sequence and the corresponding colors.



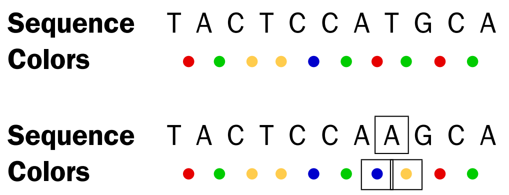
The colors do not uniquely define the sequence. Here is another sequence with the same list of colors:



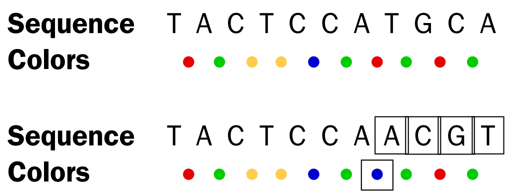
But if the first nucleotide is known, the colors do uniquely define the remaining sequence. This is exactly the strategy used in SOLiD sequencing: The first nucleotide is known from the primer used, and the remaining nucleotides are deduced from the colors.

19.8.2 Error modes

As with other sequencing technologies, errors do occur with the SOLiD technology. If a single nucleotide is changed, two colors are affected since a single nucleotide is contained in two overlapping dinucleotides:


Sequence T A C T C C A T G C A
Colors
Sequence T A C T C C A A G C A
Colors

Sometimes, a wrong color is determined at a given position. Due to the dependence between dinucleotides and colors, this affects the remaining sequence from the point of the error:


Sequence T A C T C C A T G C A
Colors
Sequence T A C T C C A A C G T
Colors

Thus, when the instrument makes an error while determining a color, the error mode is very different from when a single nucleotide is changed. This ability to differentiate different types of errors and differences is a very powerful aspect of SOLiD sequencing. With other technologies sequencing errors always appear as nucleotide differences.

19.8.3 Mapping in color space

Reads from a SOLiD sequencing run may exhibit all the same differences to a reference sequence as reads from other technologies: mismatches, insertions and deletions. On top if this, SOLiD reads may exhibit color errors, where a color is read wrongly and the rest of the read is affected. If such an error is detected, it can be corrected and the rest of the read can be converted to what it would have been without the error.

Consider this SOLiD read:

```
Read     T A C T C C A A C G T
Colors    🔴🟢🟡🟡🔵🟢🔵🟢🔴🟢
```

The first nucleotide (T) is from the primer, so is ignored in the following analysis. Now, assume that a reference sequence is this:

```
Reference  G C A C T G C A T G C A C
Colors      🔴🟢🟢🟡🟢🔴🟢🔴🟢🔴🟢🟢
```

Here, the colors are just inferred since they are not the result of a sequencing experiment.

Looking at the colors, a possible alignment presents itself:

```
Reference  G C A C T G C A T G C A C
Colors      🔴 🟢|🟢|🟡|🟢:🔴|🟢|🔴:🟢:🔴:🟢:🟢
                | | | : | | : : : :
Read           A C T C C A A C G T
Colors          🟢🟡🟡🔵🟢🔵🟢🔴🟢
```

In the beginning of the read, the nucleotides match (ACT), then there is a mismatch (G in reference and C in read), then two more matches (CA), and finally the rest of the read does not match. But, the colors match at the end of the read. So a possible interpretation of the alignment is that there is a nucleotide change in position four of the read and a color space error between positions six and seven in the read. Such an interpretation can be represented as:

```
Reference  G C A C T G C A T G C A C
               | | | : | | | | | |
Read           A C T C C A*T G C A
```

Here, the * represents a color error. The remaining part of the displayed read sequence has been adjusted according to the inferred error. So this alignment scores nine times the match score minus the mismatch cost and a color error cost. This color error cost is a new parameter that is introduced when performing read mapping in color space.

Note that a color error may be inferred before the first nucleotide of a read. This is the very first color after the known primer nucleotide that is wrong, changing the whole read.

Here is an example from a set of real SOLiD data that was reference assembled by taking color space into account using ungapped global alignments.

```
444_1840_767_F3 has 1 match with a score of 35:

    1046535 GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA 1046569    reference
            |||||||||||||||||||||||||||||||||||
            GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA            reverse read

444_1840_803_F3 has 0 matches

444_1840_980_F3 has 1 match with a score of 29:

    2620828 GCACGAAAACGCCGCGTGGCTGGATGGT*CAAC*GTC 2620862    reference
```

```
            ||||||||||||||||||||||||||||*||||*|||
            GCACGAAAACGCCGCGTGGCTGGATGGT*CAAC*GTC                        read

444_1840_1046_F3 has 1 match with a score of 32:

    3673206 TT*GGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC 3673240    reference
            ||*|||||||||||||||||||||||||||||||||
            TT*GGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC            reverse read

444_1841_22_F3 has 0 matches

444_1841_213_F3 has 1 match with a score of 29:

    1593797 CTTTG*AGCGCATTGGTCAGCGTGTAATCTCCTGCA 1593831    reference
            |||||*|||||||| |||||||||||||||||||||
            CTTTG*AGCGCATTAGTCAGCGTGTAATCTCCTGCA            reverse read
```

The first alignment is a perfect match and scores 35 since the reads are all of length 35. The next alignment has two inferred color errors that each count is -3 (marked by * between residues), so the score is *35 - 2 x 3 = 29*. Notice that the read is reported as the inferred sequence taking the color errors into account. The last alignment has one color error and one mismatch giving a score of *34 - 3 - 2 = 29*, since the mismatch cost is 2.

Running the same reference assembly without allowing for color errors, the result is:

```
444_1840_767_F3 has 1 match with a score of 35:

    1046535 GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA 1046569    reference
            |||||||||||||||||||||||||||||||||||
            GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA            reverse read

444_1840_803_F3 has 0 matches

444_1840_980_F3 has 0 matches

444_1840_1046_F3 has 1 match with a score of 29:

    3673206 TTGGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC 3673240    reference
              |||||||||||||||||||||||||||||||||
            AAGGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC            reverse read

444_1841_22_F3 has 0 matches

444_1841_213_F3 has 0 matches
```

The first alignment is still a perfect match, whereas two of the other alignment now do not match since they have more than two errors. The last alignment now only scores 29 instead of 32, because two mismatches replaced the one color error above. This shows the power of including the possibility of color errors when aligning: many more matches are found.

The reference assembly program in the *CLC Genomics Workbench* does not directly support alignment in color space only, but if such an alignment was carried out, sequence 444_1841_213_F3 would have three errors, since a nucleotide mismatch leads to two color space differences. The alignment would look like this:

```
444_1841_213_F3 has 1 match with a score of 26:

    1593797 CTTTG*AGCGCATT*G*GTCAGCGTGTAATCTCCTGCA 1593831    reference
            |||||*||||||||*|*|||||||||||||||||||||
            CTTTG*AGCGCATT*G*GTCAGCGTGTAATCTCCTGCA            reverse read
```

So, the optimal solution is to both allow nucleotide mismatches and color errors in the same program when dealing with color space data. This is the approach taken by the assembly program in the *CLC Genomics Workbench*.

19.8.4 Viewing color space information

Importing data from SOLiD systems (see section 19.1.3) will from *CLC Genomics Workbench* version 3.1 be imported as color space. This means that if you open the imported data, it will look like figure 19.63

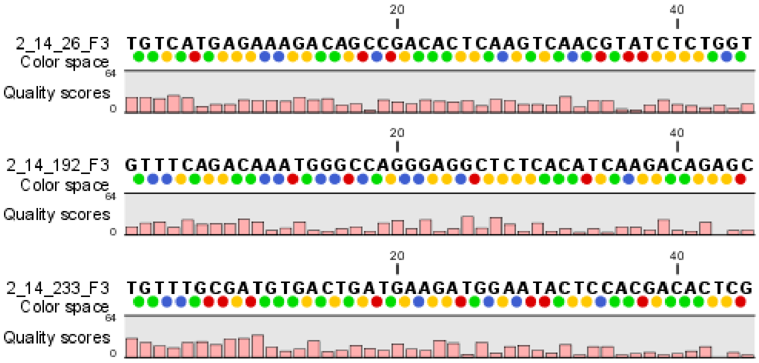

20
40
2_14_26_F3
TGTCATGAGAAAGACAGCCGACACTCAAGTCAACGTATCTCTGGT
Color space
64
Quality scores
0
20
40
2_14_192_F3
GTTTCAGACAAATGGGCCAGGGAGGCTCTCACATCAAGACAGAGC
Color space
64
Quality scores
0
20
40
2_14_233_F3
TGTTTGCGATGTGACTGATGAAGATGGAATACTCCACGACACTCG
Color space
64
Quality scores
0

Figure 19.63: *Color space sequence list.*

In the **Side Panel** under **Nucleotide info**, you find the **Color space encoding** group which lets you define a few settings for how the colors should appear. These settings are also found in the side panel of mapping results and single sequences.

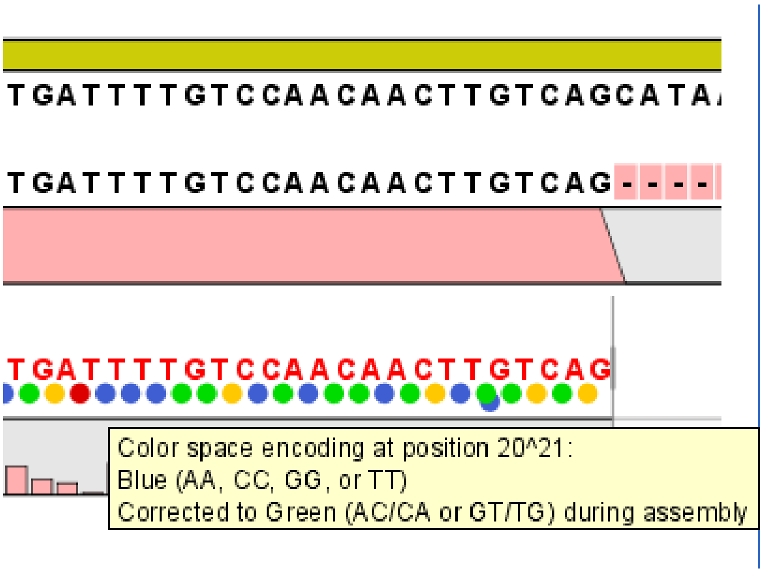
Infer encoding This is used if you want to display the colors for non-color space sequence (e.g. a reference sequence). The colors are then simply inferred from the sequence.

Show corrections This is only relevant for mapping results - it will show where the mapping process has detected color errors. An example of a color error is shown in figure 19.64.

Hide unaligned ends This option determines whether color for the unaligned ends of reads should be displayed. It also controls whether colors should be shown for gaps. The idea behind this is that these color dots will interfere with the color alignment, so it is possible to turn them off.

19.9 Interpreting genome-scale mappings

A big challenge when working with high-throughput sequencing projects is interpretation of the data. Section 19.11 describes how to automatically detect SNPs, whereas this section describes the manual inspection and interpretation techniques which are guided by visual information about



Color space encoding at position 20^21:
Blue (AA, CC, GG, or TT)
Corrected to Green (AC/CA or GT/TG) during assembly

Figure 19.64: *One of the dots have both a blue and a green color. This is because this color has been corrected during mapping. Putting the mouse on the dot displays the small explanatory message.*

the mapping. (We will not cover all the functionalities of the mapping view here, instead we refer to section 18.6 for general information about viewing and editing the resulting mappings).

Of particular interest for high-throughput sequencing data is probably the opportunity to extract part of mapping result, see section 18.6.6.

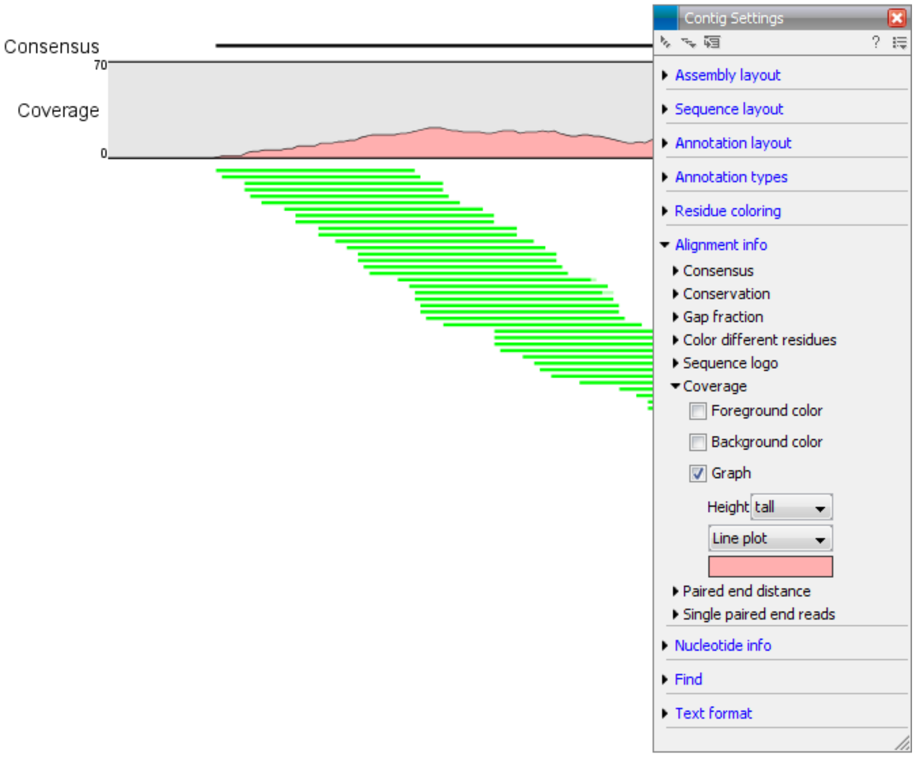
19.9.1 Getting an overview - zooming and navigating

Results from mapping high-throughput sequencing data may be extremely large, requiring an extra effort when you navigate and zoom the view. Besides the normal zoom tools and scrolling via the arrow keys, there are some of the settings in the **Side Panel** which can help you navigate a large mapping:

- **Gather sequences at top**. Under **Read layout** at the top of the Side Panel. you find this option. When you zoom in, only the reads aligning to the visible part of the view will be shown. This will save a lot of vertical scrolling.
- **Compactness**. Under **Sequence layout**, you can use different modes of compactness. This affects the way reads are shown. For example, you can display reads as **Compact** - very thin stacked lines as shown in figure 19.65. The compactness also affects what information should be displayed below the reads (i.e. quality scores or chromatogram traces).
- **Text size**. Under **Text format** at the bottom of the **Side Panel**, you decrease the size of the text. This can improve the overview of the results (at the expense of legibility of sequence names etc.).

19.9.2 Single reads - coverage and conflicts

When you only have single reads data, **coverage** is one of the main resources for interpretation. You can display a coverage graph by clicking the checkbox in the Side Panel as shown in figure 19.65.



□ Foreground color

□ Background color

☑ Graph

Height tall

Line plot

Figure 19.65: *The coverage graph can be displayed in the Side Panel under Alignment info.*

If you wish to see the exact coverage at a certain position, place the mouse cursor on the graph and see the exact value in the status bar at the very lower right corner of the Workbench window. Learn how to export the data behind the graph in section 7.4.

When you zoom out on a large reference sequence, it may be difficult to discern smaller regions of low coverage. In this case, click the **Find Low Coverage** button at the top of the **Side Panel**. Clicking once will select the first part of the mapping with coverage at or below the number specified above the button (**Low coverage threshold**). Click again to find the next part with low coverage.

When mapping reads to a reference, a region of no coverage indicates genome-scale mutations. If the sequencing data contains e.g. a deletion, this will appear as a region of no coverage.

Problems during the sequencing process will also result in low coverage regions. In this case, you may wish to re-sequence these parts, e.g. using traditional "Sanger"-sequencing techniques. Due to the integrated nature of the *CLC Genomics Workbench* you can easily go to the primer designer and design PCR and sequencing primers to cover the low-coverage region. First select the low coverage region (and some extra nucleotides in order to get a good quality of the sequencing in the area of interest), and then:

> **right-click the selection | Open Selection in New View () | Show Primer Design () at the bottom of the view**

Read more about designing primers in section 17.

Besides looking at coverage, you can of course also inspect the conflicts by clicking the **Find Conflict** button at the top of the **Side Panel**. However, this will be practically impossible for large mappings, and it will not provide the same kind of overview as other approaches.

19.9.3 Interpreting genomic re-arrangements

Most of the analyses in this section are based on paired data which allows for much more powerful approaches to detecting genome rearrangements. Figure 19.66 shows a part of a mapping with paired reads.

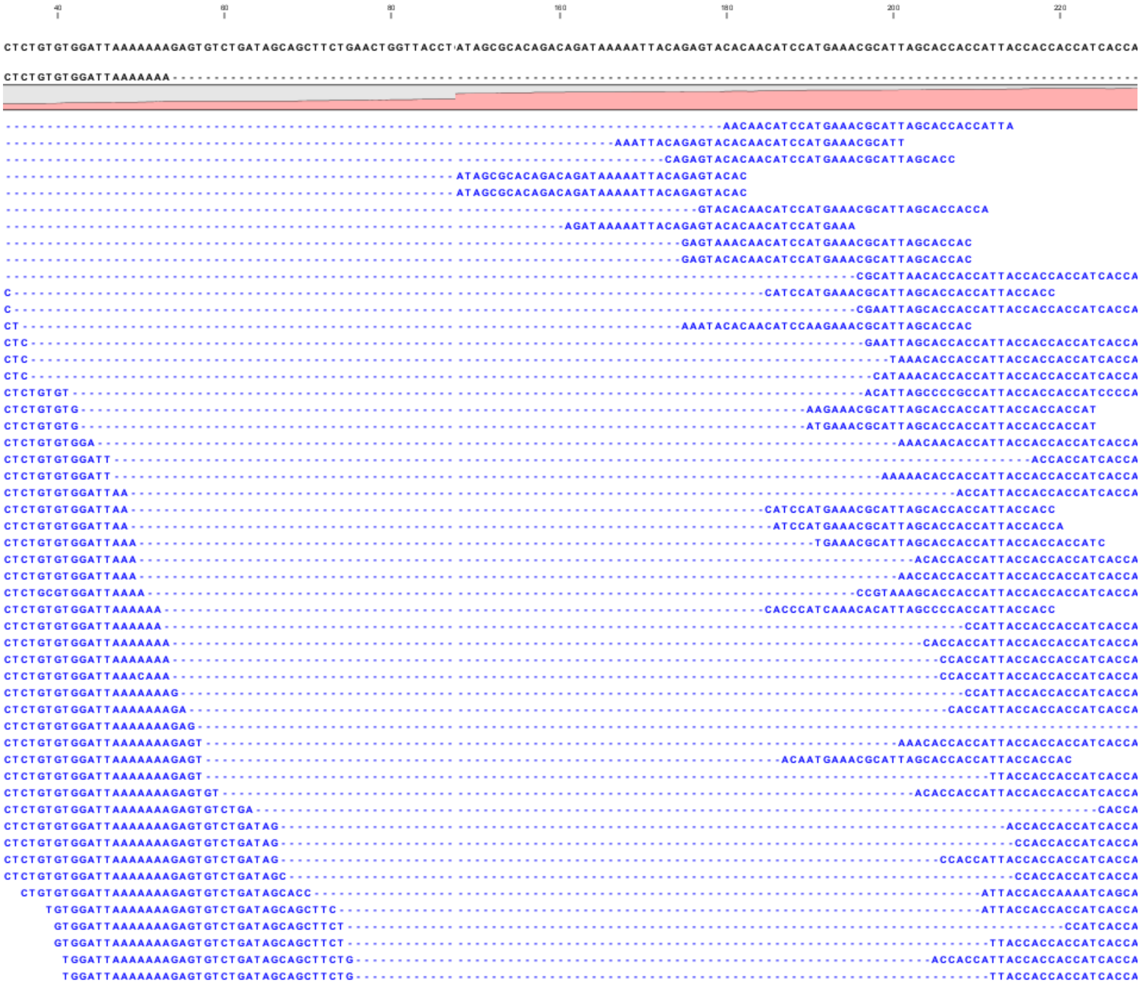


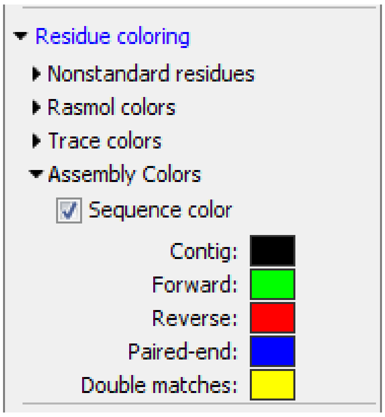
Figure 19.66: *Paired reads are shown with both sequences in the pair on the same line. The letters are probably too small to read, but it gives you the impression of how it looks.*

You can see that the sequences are colored blue and this leads us to the color settings in the **Side Panel**: under **Residue coloring** you find the group **Sequence colors** where you can specify the following colors:

- **Mapping**. The color of the consensus and reference sequence. Black per default.
- **Forward**. The color of forward reads (single reads). Green per default.
- **Reverse**. The color of reverse reads (single reads). Red per default.
- **Paired**. The color of paired reads. Blue per default.
- **Non-specific matches**. When a read would have matched another place in the mapping, it is considered a double match. This color will "overrule" the other colors. Note that if your

mapping with several reference sequences, either using *de novo* assembly or read mapping with multiple reference sequences, a read is considered a double match when it matches more than once *across all the contigs/references*. A double match is yellow per default.

The settings are shown in figure 19.67.



▾ Residue coloring
▸ Nonstandard residues
▸ Rasmol colors
▸ Trace colors
▾ Assembly Colors
☑ Sequence color
Contig: ⬛
Forward: 🟩
Reverse: 🟥
Paired-end: 🟦
Double matches: 🟨

Figure 19.67: *Coloring of the reads.*

In addition to these colors, there are three graphs that will prove helpful when inspecting the paired reads, both found under **Alignment info** in the **Side Panel** (see figure 19.68):

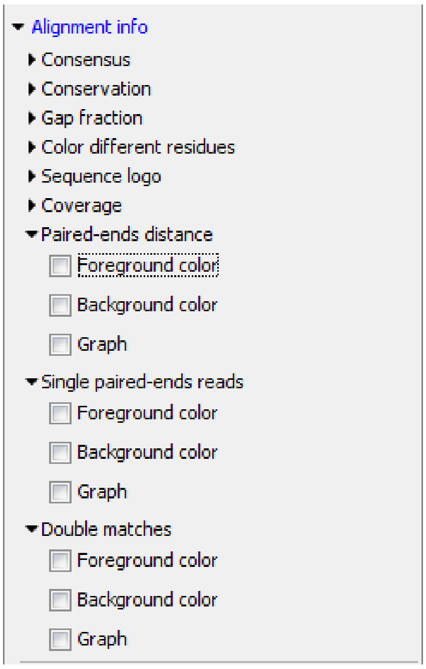
- **Paired distance**. Displays the average distance between the forward and the reverse read in a pair.
- **Single paired reads**. Displays the percentage of the reads where only one of the reads in a pair matches.
- **Non-perfect matches**. Displays the percentage of the reads covering the current position which have at least one mismatch or a gap (the mismatch or gap does not need to be on this position - if there is just one anywhere on the read, it will count).
- **Non-specific matches**. Displays the percentage of the reads which match more than once. Note that if you are mapping against several sequences, either using *de novo* assembly or read mapping with multiple reference sequences, a read is considered a non-sepcific match when it matches more than once *across all the contigs/references*. A non-specific match is yellow per default.

These three graphs in combination with the read colors provide a great deal of information, guiding interpretations of the mapping result. A few examples will give directions on how to take advantage of these powerful tools:

Insertions

Looking at the **Single paired reads** graph in figure 19.69, you can see a sudden rise and fall. This means that at this position, only one part of the pair matches the reference sequence.

Zooming in on the reads, you see how the color of the reads changes (see figure 19.70. They go from blue (paired) to green, meaning that at this point, the reverse part of the paired reads no longer match the reference sequence.



- Alignment info
  - Consensus
  - Conservation
  - Gap fraction
  - Color different residues
  - Sequence logo
  - Coverage
  - Paired-ends distance
    - □ Foreground color
    - □ Background color
    - □ Graph
  - Single paired-ends reads
    - □ Foreground color
    - □ Background color
    - □ Graph
  - Double matches
    - □ Foreground color
    - □ Background color
    - □ Graph

Figure 19.68: *More information about paired reads can be displayed in the Side Panel.*

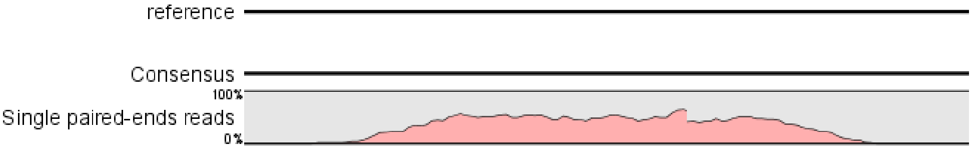


Figure 19.69: *More information about paired reads can be displayed in the Side Panel.*

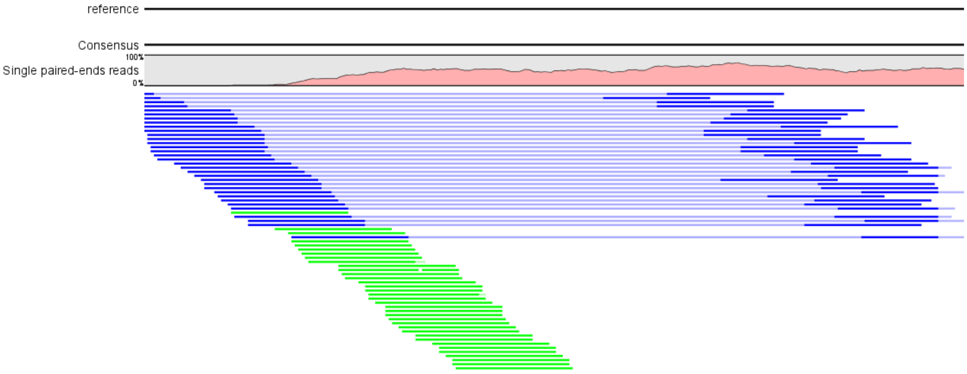


Figure 19.70: *Zooming where the single reads kick in.*

Since their reverse partners do not match the reference, there must be an insertion in the sequenced data. Looking further down the view, the color changes from green to a combination of red (only reverse reads match) and blue (see figure 19.71).

The reverse reads colored in red have a forward counterpart which do not match the reference sequence, for the same reason as we see the lonely forward reads before the insertion. Among the reverse reads, the "ordinary" paired reads start again, marking the end of the insertion.

As we now have established the presence of an insertion, it would be nice to know the exact location of insertion. You can see its exact position in figure 19.71 where the green reads stop matching the reference and the reverse reads take over (marked by the vertical red selection

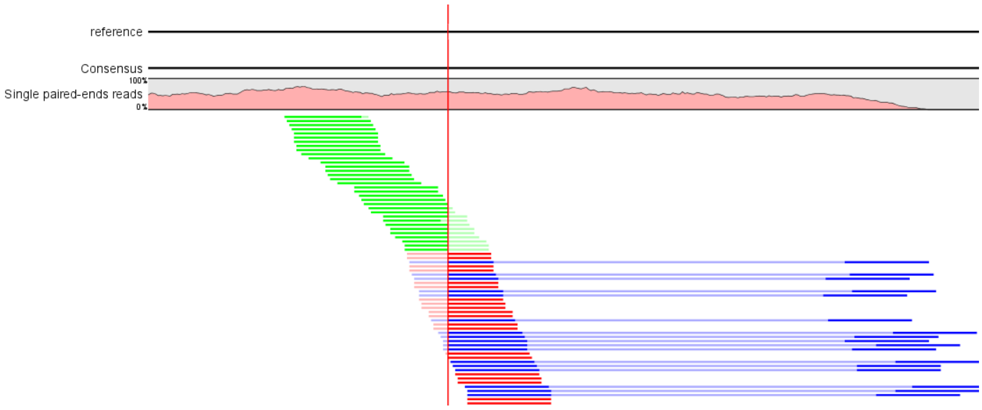


Figure 19.71: *Zooming where the paired reads kick in again.*

line).

Deletions

Deletions are much easier to detect - they are simply areas of no coverage (see figure 19.72).

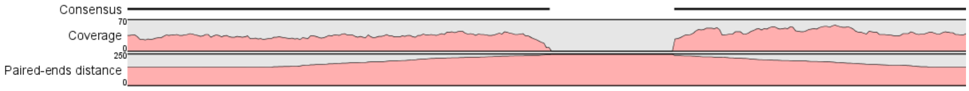


Figure 19.72: *A deletion in the sequenced data results in coverage of 0.*

Depending on the size of the deletion, you will see a rise in other graphs as well:

- A small deletion will result in an increase of the **Paired distance**, because the gap between the forward and the reverse read will just extend the deletion. This is the case in figure 19.72.
- A larger deletion will result in an increase of **Single paired reads** when the deletion is larger than the maximum distance allowed between paired reads (because the "other" part of the read has a match which is too far away). This maximum value can be changed when mapping the reads, see section 19.5. This is not illustrated.

When you zoom in on the deletion, you can see how the distance between the reads increase (see figure 19.73).

Duplications

In figure 19.74, the **Non-specific matches** graph is now shown.

The Non-specific matches are reads that match more than once on the reference sequence. Zooming in on the reads puts a new color into play as shown in figure 19.75.

The yellow color means that the reads also match other positions on the reference, and this indicates that there is a duplication.

For a smaller duplication, you will see an increase in the **Paired distance**, because some of the reads are then matched to the other part of the duplication (this is shown in figure 19.76.

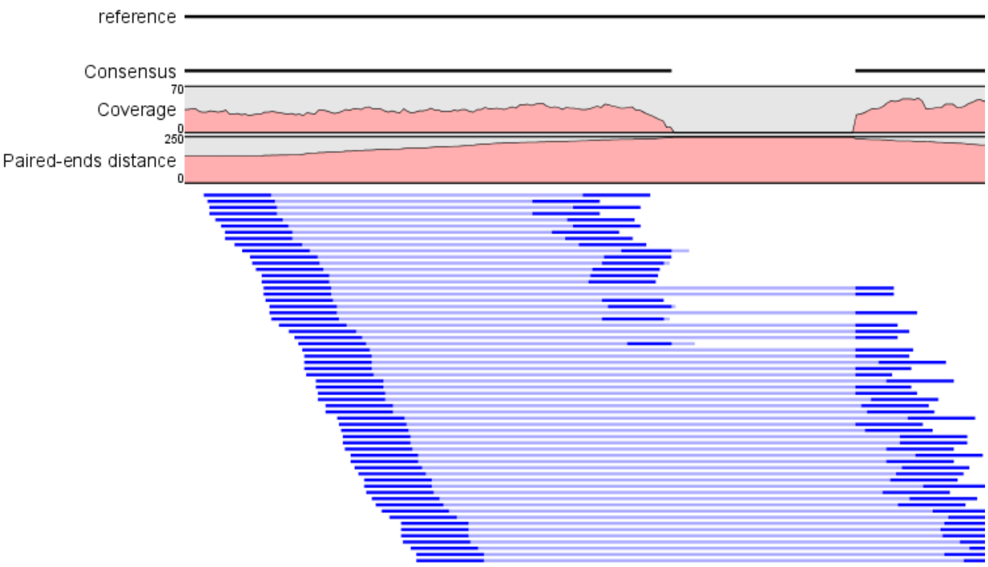


Figure 19.73: *Each part of the pair still match because the deletion is smaller than the maximum distance between the reads.*

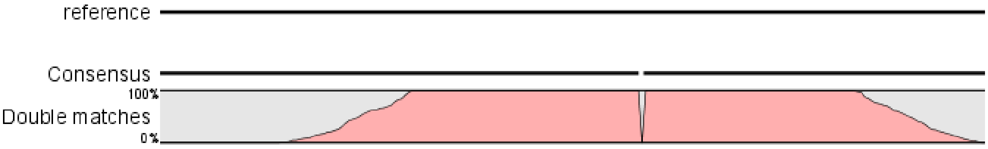


Figure 19.74: *A rise in the Non-specific matches.*

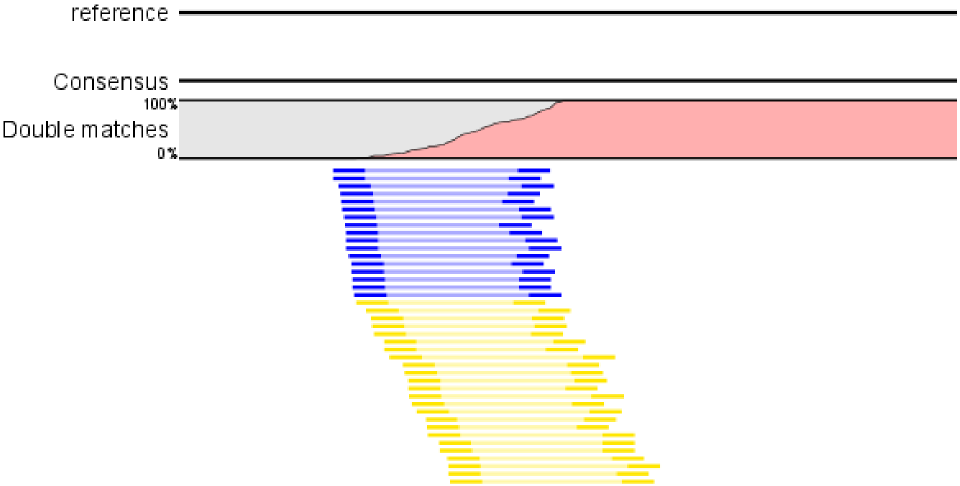


Figure 19.75: *Non-specific matches are shown in yellow.*

Inversions

The interesting part in figure 19.77 is once again the **Single paired reads** graph which display a distinct pattern.

The explanation of this is as follows. When the first peak starts, it is because the reverse part of the pairs no longer matches the reference sequence. This is shown in detail in figure 19.78.

Scrolling further along the view we can see the starting point of the inverted region. This is were the forward reads ends. At the same point, you will see a new pattern: a combination of reverse

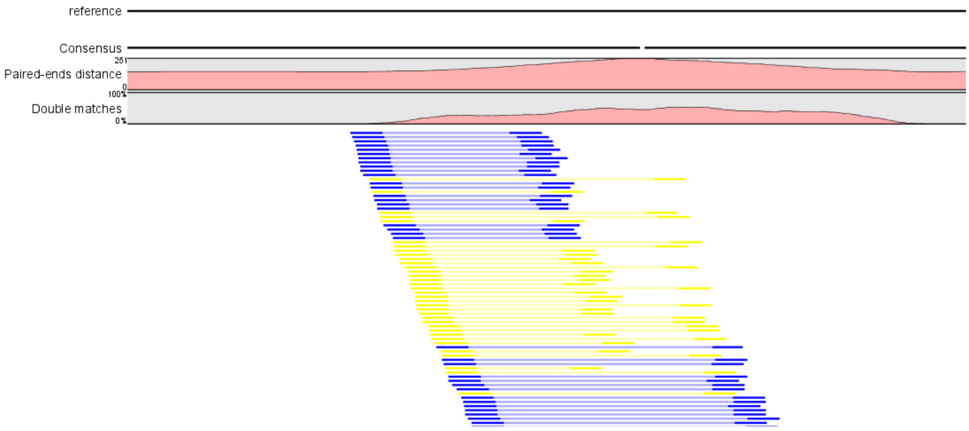


Figure 19.76: *Paired distance increases.*

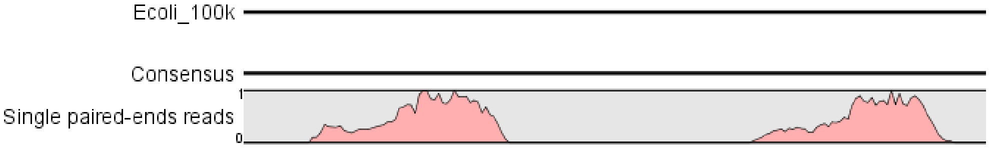


Figure 19.77: *Two peaks in the Single paired reads graph.*

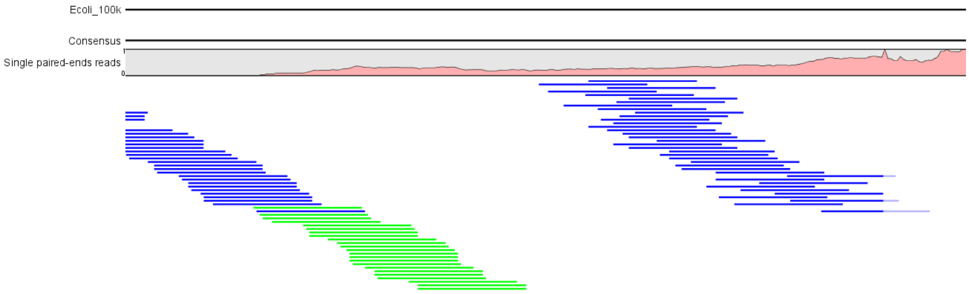


Figure 19.78: *Just before the inversion, only the forward reads match.*

and paired reads as shown in figure 19.79.

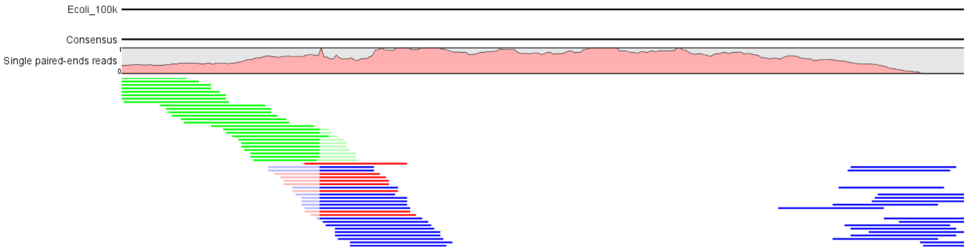


Figure 19.79: *The inversion starts where the reads shift from green (forward) to a combination of red and blue (reverse and paired) reads.*

The forward counterpart of the reverse reads has no match because of the inversion, whereas the paired reads have been reversed compared to the other paired reads in the mapping (this is not visible in the user interface, but a conclusion you can draw from the pattern of the other reads). Scrolling to the end of the inversion, you will see a similar pattern as in the beginning - it is just mirrored: Forward reads kick in at the end of the inversion, and reverse reads take over at when we get back to a "normal" sequence (see figure 19.80).

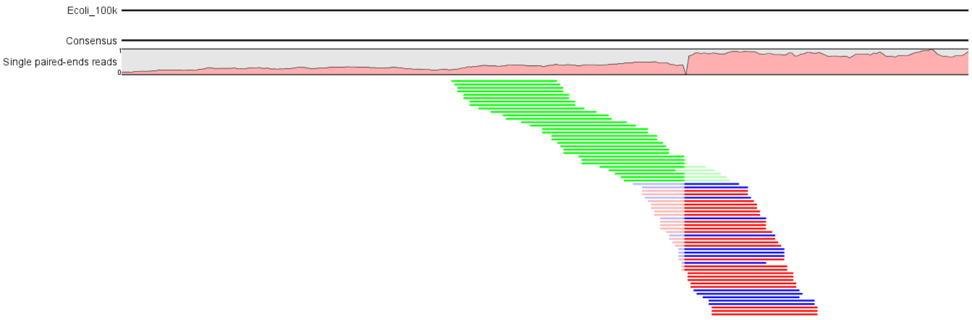


Figure 19.80: *The inversion ends where the reads shift from green (forward) to a combination of red and blue (reverse and paired) reads.*

19.9.4 Further analysis of read mappings

When you have finished interpreting the results, you may wish to perform additional analyses on the sequenced data. Apart from SNP detection, described in section 19.11, you can use the consensus or reference sequence for all the other analyses of the *CLC Genomics Workbench*. To get started on using the consensus or reference sequences, you need to know a little trick:

> **right-click the name of the consensus/reference sequence | Open This Sequence in New View ( )**

Alternatively, if you have several mappings in a table (as described in sections 19.4.5 and 19.5.5), you can extract the consensus sequences by selecting the relevant rows by pressing **Open Consensus** at the bottom of the view.

In any case, the sequence(s) will now have a life of their own and are no longer attached to the original mapping.

If you have annotated open reading frames on the sequence and wish to analyze each of these regions separately, e.g. translating and BLASTing or using other protein analysis tools, you can extract all the ORF annotations by using our **Extract Annotations** plug-in, available from the **Plug-in Manager** ( ). This will give you a sequence list containing all the ORFs, making it easy to do batch analyses with other tools from *CLC Genomics Workbench*.

19.10 Merge mapping results

If you have performed two mappings with the same reference sequences, you can merge the results using the **Merge Mapping Results** ( ). This can be useful in situations where you have already performed a mapping with one data set, and you receive a second data set that you want to have mapped together with the first one. In this case, you can run a new mapping of the second data set and merge the results:

> **Toolbox | High-throughput Sequencing ( ) | Merge Mapping Results ( )**

This opens a dialog where you can select two or more mapping results. Note that they have to be based on the same reference sequences (it doesn't have to be the same file, but the sequence (the residues) should be identical).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. . For all the mappings that could be merged, a new mapping will be created. Even if you have used a mapping table as input, the result will be individual mappings. Note that the consensus sequence is updated to reflect the merge. The consensus voting scheme for the first contig is used to determine the consensus sequence. This also means that for large mappings, the data processing can be quite demanding for your computer.

19.11 SNP detection

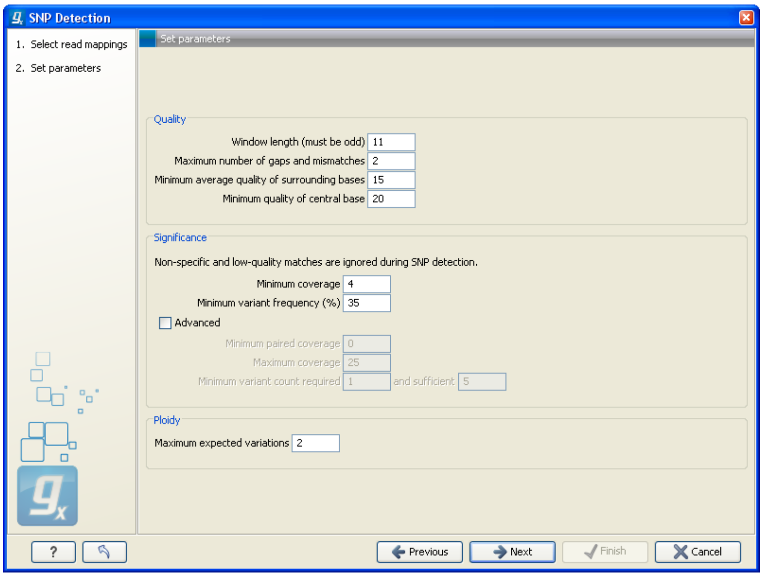
Instead of manually checking all the conflicts of a mapping to discover significant single-nucleotide variations, *CLC Genomics Workbench* offers automated SNP detection (see our Bioinformatics explained article on SNPs at `http://www.clcbio.com/BE`). The SNP detection in *CLC Genomics Workbench* is based on the *Neighborhood Quality Standard (NQS)* algorithm of [Altshuler et al., 2000] (also see [Brockman et al., 2008] for more information).

Based on your specifications on what you consider a valid SNP, the SNP detection will scan through the entire data and report all the SNPs that meet the requirements:

Toolbox | High-throughput Sequencing () | SNP detection ()

This opens a dialog where you can select read mappings ()/ () to scan for SNPs (see sections 19.4 and 19.5 for information on how to map reads). You can also select RNA-Seq results () as input.

Clicking **Next** will display the dialog shown in figure 19.81



Quality
Window length (must be odd) 11
Maximum number of gaps and mismatches 2
Minimum average quality of surrounding bases 15
Minimum quality of central base 20

Significance
Non-specific and low-quality matches are ignored during SNP detection.
Minimum coverage 4
Minimum variant frequency (%) 35
☐ Advanced
Minimum paired coverage 0
Maximum coverage 25
Minimum variant count required 1 and sufficient 5

Ploidy
Maximum expected variations 2

Figure 19.81: *SNP detection parameters.*

19.11.1 Assessing the quality of the neighborhood bases

The SNP detection will look at each position in the mapping to determine if there is a SNP at this position. In order to make a qualified assessment, it also considers the general quality of the neighboring bases. The **Window size** is used to determine how far away from the current position this quality assessment should extend, and it can be specified in the upper part of the dialog. Note that at the ends of the read, an asymmetric window of the specified length is used.

If the mapping is based on local alignment of the reads, there will be some reads with un-aligned ends (these ends are faded when you look at the mapping). These unaligned ends are not included in the scanning for SNPs but they are included in the quality filtering (elaborated below).

In figure 19.82, you can see an example with a window size of 11. The current position is high-lighted, and the horizontal high-lighting marks the nucleotides considered for a read when using a window size of 11.

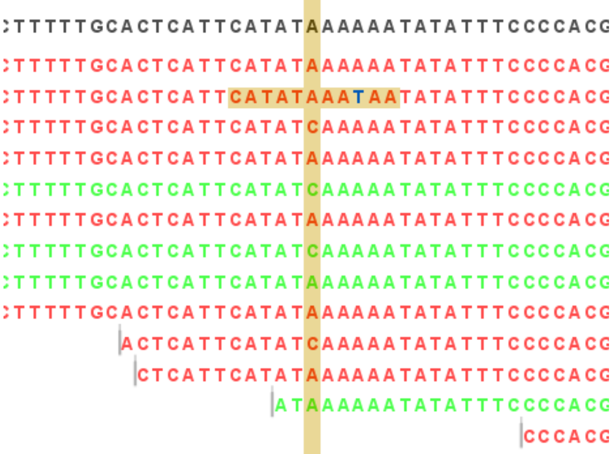

:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAATAATATATTTCCCCACG
:TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
:TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
ACTCATTCATATCAAAAATATATTTCCCCACG
CTCATTCATATAAAAAATATATTTCCCCACG
ATAAAAAATATATTTCCCCACG
CCCACG

Figure 19.82: *An example of a window size of 11 nucleotides.*

For each read and within the given window size,[1] the following two parameters are used to assess the quality:

- **Minimum average quality of surrounding bases.** The average quality score of the nucleotides in a read within the specified window length has to exceed this threshold for the base to be included in the SNP calculation for this position (learn more about importing quality scores from different sequencing platforms in section 19.1).
- **Max. number of gaps and mismatches.** The number of gaps and mismatches allowed within the window length of the read. If there are more gaps or mismatches, this read will not be included in the SNP calculation at this position. Unaligned regions (the faded parts of a read) also count as mismatches, *even if some of the bases match*.

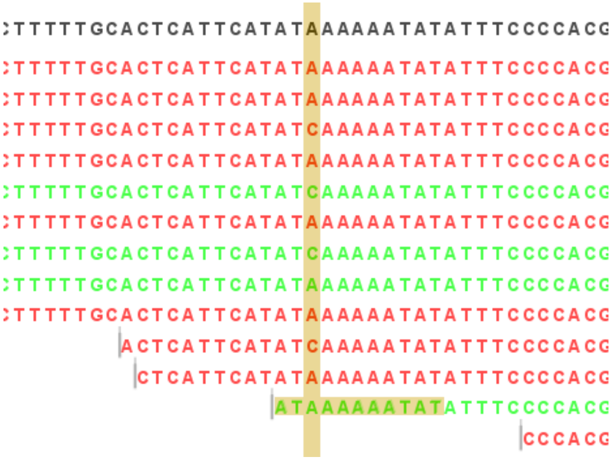
Note that for sequences without quality scores, the quality score settings will have no effect. In this case only the gap/mismatch threshold will be used for filtering low quality reads.

Figure 19.82 shows an example of a read with a mismatch, marked in dark blue. The mismatch is inside the window of 11 nucleotides.

When looking at a position near the end of a read (like the read at the bottom in figure 19.82), the window will be asymmetric as shown in figure 19.83. The window size will thus still be 11 in this case.

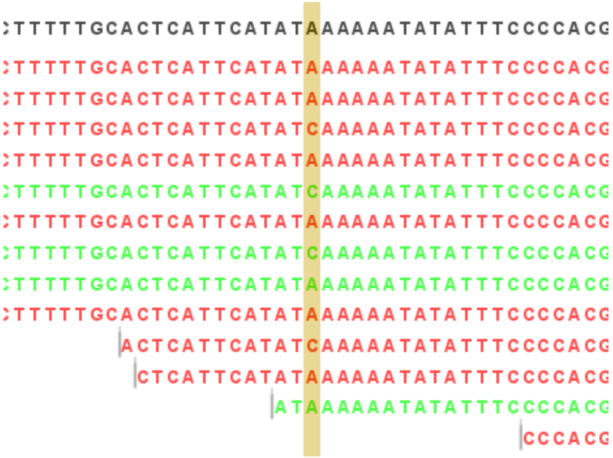
Besides looking horizontally within a window for each read, the quality of the central base is also examined: **Minimum quality of central base**. This is the quality score for the central base, i.e. the bases in the column high-lighted in figure 19.84. Bases with a quality score below this value are not considered in the SNP calculation at this position.

[1] The window size is defined as the number of positions in the local alignment between that particular read and the reference sequence (for de novo assembly it would be the consensus sequence).)



```
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG

TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
       ACTCATTCATATCAAAAATATATTTCCCCACG
        CTCATTCATATAAAAAATATATTTCCCCACG
                 ATAAAAAATATATTTCCCCACG
                                 CCCACG
```

Figure 19.83: *A window near the end of a read.*



```
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG

TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATCAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
TTTTTGCACTCATTCATATAAAAAATATATTTCCCCACG
       ACTCATTCATATCAAAAATATATTTCCCCACG
        CTCATTCATATAAAAAATATATTTCCCCACG
                 ATAAAAAATATATTTCCCCACG
                                 CCCACG
```

Figure 19.84: *A column of central bases in the neighborhood.*

In addition to low-quality reads, reads that match more than once on the reference sequence(s) are also ignored. These reads are also called **Non-specific matches** and are colored in yellow in the view. Reads from broken pairs are also ignored (regardless of the **Minimum paired coverage** setting, see below).

19.11.2 Significance of variation: is it a SNP?

At a given position, when the reads with low quality and multiple matches have been removed, the reads which pass the quality assessment will be compared to the reference sequence to see if they are different at this position (for *de novo* assembly the consensus sequence is used for comparison). For a variation to count as a SNP, it has to comply with the significance threshold specified in the dialog shown in figure 19.81.

- **Minimum coverage**. If SNPs were called in areas of low coverage, you would get a higher amount of false positives. Therefore you can set the minimum coverage for a SNP to be called. Note that the coverage is counted as the number of valid reads at the current position (i.e. the reads remaining when the quality assessment has filtered out the bad ones).
- **Minimum variant frequency**. This option is the threshold for the number of reads that

display a variant at a given position. The threshold can be set as a frequency percentage or as a count. Setting the percentage at 35 % means that at least 35 % of the validated reads at this position should have a different base.

Below, there is an **Advanced** option letting you specify additional requirements. These will only take effect if the **Advanced** checkbox is checked.

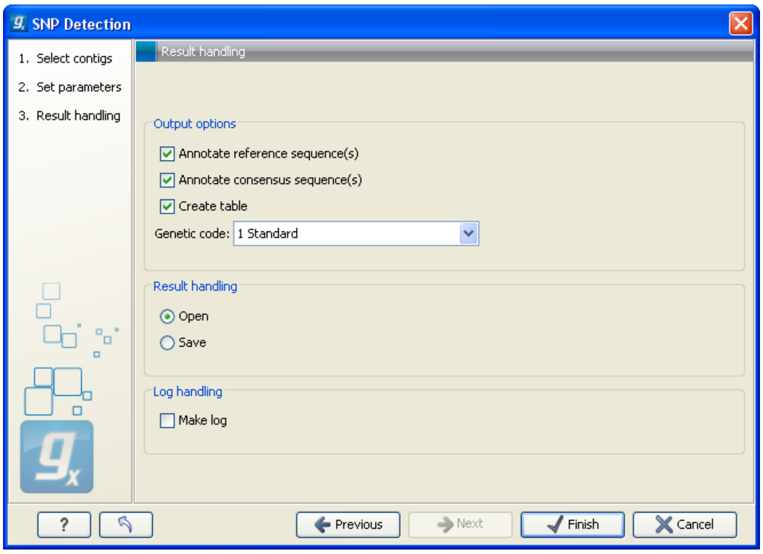
- **Minimum paired coverage**. In samples based on paired data, more confidence is often attributed to valid paired reads than to single reads. You can therefore set the minimum coverage of valid paired reads in addition to the minimum coverage of all reads. Again, the paired coverage is counted as the number of valid reads completely covering the SNP (the space between mating pairs does not cover anything.) Note that regardless of this setting, reads from broken pairs are never considered for DIP detection.
- **Maximum coverage**. Although it sounds counter-intuitive at first, there is also a good reason to be suspicious about high-coverage regions. Read coverage often displays peaks in repetitive regions where the alignment is not very trustworthy. Setting the maximum coverage threshold a little higher than the expected average coverage (allowing for some variation in coverage) can be helpful in ruling out false positives from such regions. You can see the distribution of coverage by creating a detailed mapping report (see section 19.6.1). The result table created by the SNP detection includes information about coverage, so you can specify a high threshold in this dialog, check the coverage in the result afterwards, and then run the SNP detection again with an adjusted threshold.
- **Minimum variant count**. This option is the threshold for the number of reads that display a variant at a given position. In addition to the percentage setting in the simple panel above, these settings are based on absolute counts. If the count required is set to 3, and the sufficient count is set to 5, it means that **even though** less than the required percentage of the reads have a variant base, it will still be reported as a SNP if at least 5 reads have it. However, if the count is 2, the SNP will not be called, regardless the percentage setting. This distinction is especially useful with deep sequencing data where you have very high coverage and many different alleles. In this case, the percentage threshold is not suitable for finding valid SNPs in a small subset of the data. If you are not interested in reporting SNPs based on counts but only rely on the relative frequency, you can simply set the sufficient count number very high.

Positions where the reference sequences (consensus sequences for de novo assembly) have gaps and unaligned ends of the reads (faded part of the read) will not be considered in the SNP detection.

The last setting in this dialog (figure 19.81) concerns ploidy: **Maximum expected variations**. This is not a filtering option but a reporting option that is related to the minimum variant frequency setting. If the frequency or count threshold is set low enough the algorithm can call more allelic variants than the ploidy number of the organism sequenced. Note here, that the number of allelic variants is the variants found within both the sequencing reads and the reference sequence. Such a result may occur as a real result but is inconsistent with the common assumption of an infinite sites mutation model where mutations are assumed to be so rare that they never affect the same position twice. For this reason, you can use the maximum expected variations setting to mark reported SNPs as "complex" when they involve more allelic variations then expected from the ploidy number under an infinite sites model. Note, that with this interpretation the "complex" flag holds true regardless of whether the sequencing data are generated from a population sample or from an individual sample (however, see below for an exception). For example, using a minimum variant frequency of 30% with a diploid organism, you are allowing SNPs with up to 3 variations within the sequencing reads, and by then setting the maximum expected variations count to 2 (the default), any SNPs with 3 variations will be marked as "complex" (see below). A ploidy level of 1 with two allelic variants represents a special case. Two allelic variants can occur if all reads are found to agree on one base that differs from the reference. Here, the number of allelic variants is higher than the ploidy level. but this is not inconsistent with an infinite sites mutation model and will not be termed complex. Two allelic variants can also occur if two variants are found within the sequencing reads where one of the variants is the same as the reference. Again, the data are not inconsistent with an infinite sites model if the sequencing data are generated from a population sample, but they are inconsistent with a clonal mutation-free origin of a sample from a single individual. For this reason we have chosen to also designate this latter case as "complex".

19.11.3 Reporting the SNPs

When you click **Next**, you will be able to specify how the SNPs should be reported (see figure 19.85).



Output options

☑ Annotate reference sequence(s)

☑ Annotate consensus sequence(s)

☑ Create table

Genetic code: 1 Standard

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 19.85: *Reporting options for SNP detection.*

- **Add SNP annotations to reference**. This will add an annotation for each SNP to the reference sequence.
- **Add SNP annotations to consensus**. This will add an annotation for each SNP to the consensus sequence.
- **Create table**. This will create a table showing all the SNPs found in the data set. The table will provide a valuable overview, whereas the annotations are useful for detailed inspection of a SNP, and also if the consensus sequence is used in further analysis in the *CLC Genomics Workbench*. The table displays the same information as the annotation for each SNP.

- **Genetic code.** When reporting the effect of a SNP on the amino acid, this translation table specified here is used.

Figure 19.86 shows a SNP annotation.

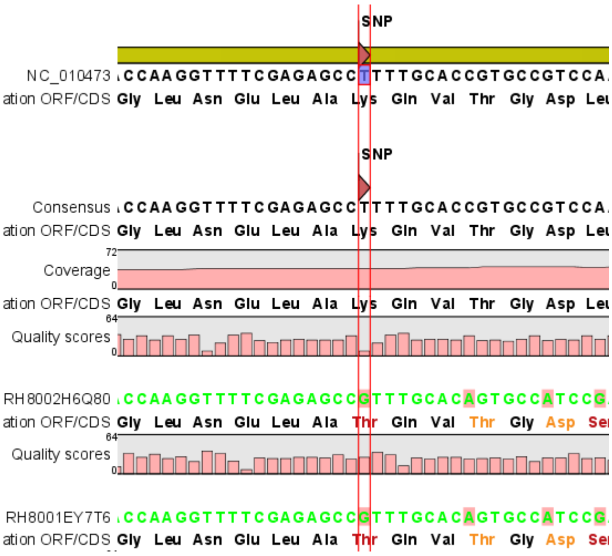
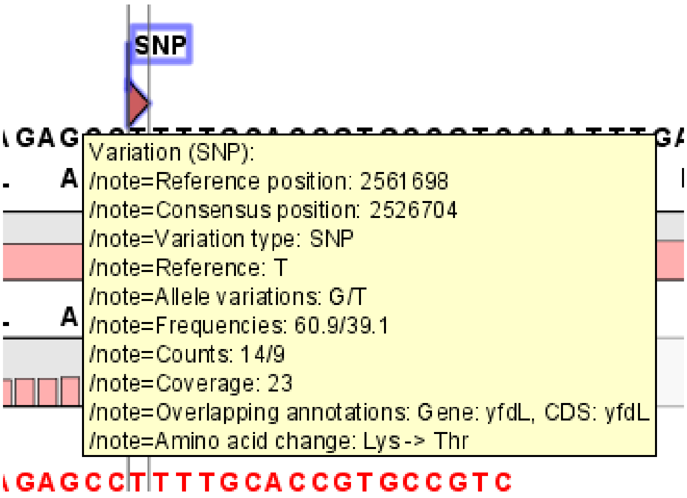

SNP
NC_010473 CCAAGGTTTTCGAGAGCCTTTTGCACCGTGCCGTCCA
ation ORF/CDS Gly Leu Asn Glu Leu Ala Lys Gln Val Thr Gly Asp Leu
SNP
Consensus CCAAGGTTTTCGAGAGCCTTTTGCACCGTGCCGTCCA
ation ORF/CDS Gly Leu Asn Glu Leu Ala Lys Gln Val Thr Gly Asp Leu
72
Coverage
0
ation ORF/CDS Gly Leu Asn Glu Leu Ala Lys Gln Val Thr Gly Asp Leu
64
Quality scores
0
RH8002H6Q80 CCAAGGTTTTCGAGAGCCGTTTGCACAGTGCCATCCG
ation ORF/CDS Gly Leu Asn Glu Leu Ala Thr Gln Val Thr Gly Asp Ser
64
Quality scores
0
RH8001EY7T6 CCAAGGTTTTCGAGAGCCGTTTGCACAGTGCCATCCG
ation ORF/CDS Gly Leu Asn Glu Leu Ala Thr Gln Val Thr Gly Asp Ser

Figure 19.86: *A SNP annotation within a coding region.*

The SNP in figure 19.86 is within a coding region and you can see that one of the variations actually changes the protein product (from Lys to Thr). Placing your mouse on the annotation will reveal additional information about the SNP as shown in figure 19.87.



Variation (SNP):
/note=Reference position: 2561698
/note=Consensus position: 2526704
/note=Variation type: SNP
/note=Reference: T
/note=Allele variations: G/T
/note=Frequencies: 60.9/39.1
/note=Counts: 14/9
/note=Coverage: 23
/note=Overlapping annotations: Gene: yfdL, CDS: yfdL
/note=Amino acid change: Lys-> Thr

Figure 19.87: *A SNP annotation with associated information.*

The SNP annotation includes the following additional information:

- **Reference position.** The SNP's position on the reference sequence.
- **Consensus position.** The SNP's position on the consensus sequence.
- **Variation type.** The SNP is described as complex, if it has more variations than specified in the ploidy setting in figure 19.81.

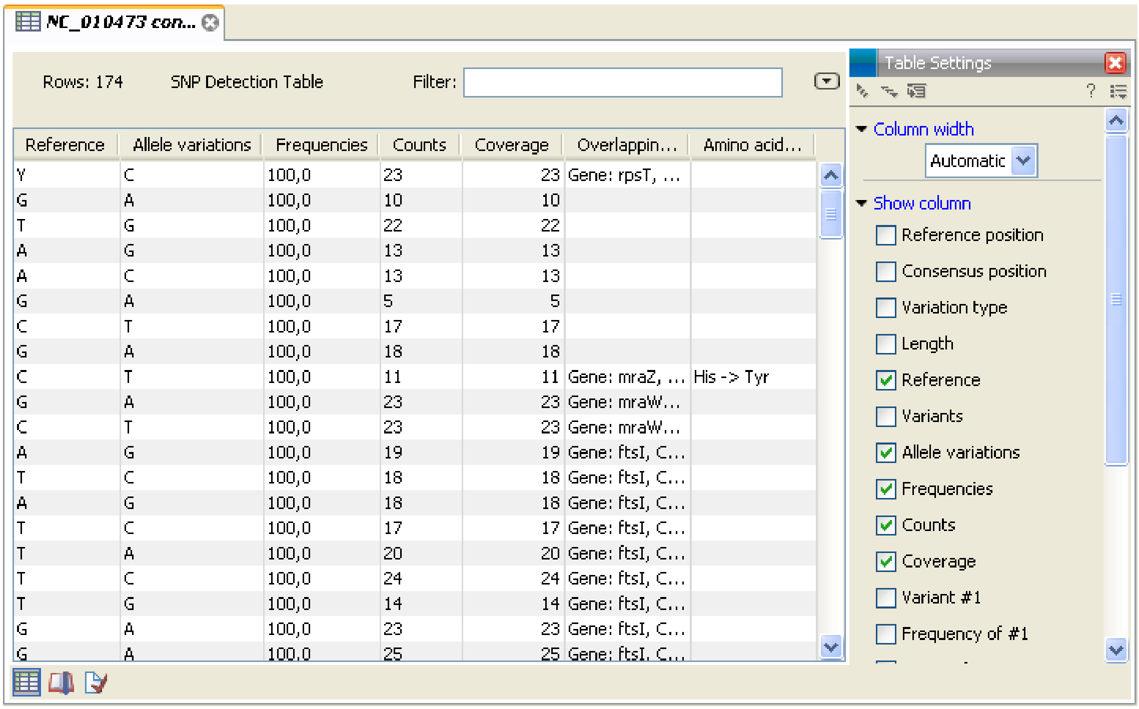
- **Length**. The length of the SNP will always be one, as the name implies, unless two SNPs are found within the same codon (see section 19.11.4).
- **Reference**. The base found in the reference sequence. For results from de novo assembly, it will be the base found in the consensus sequence.
- **Variants**. The number of variants among the reads.
- **Allele variations**. Displays which bases are found at this position. In the example shown in figure 19.86, the reference sequence has a **T** whereas some of the reads have a **G**.
- **Frequencies**. The frequency of a given variant. In the example shown in figures 19.86 and 19.87, 61 % of the reads have a **G** and 39 % have a **T**.
- **Counts**. This is similar to the frequency just reported in absolte numbers. In the example shown in figures 19.86 and 19.87, 14 reads have a **G** and 9 have a **T**.
- **Coverage**. The coverage at the SNP position. Note that only the reads that pass the quality filter will be reported here.
- **Variant numbers and frequencies**. The information from the Allele variations, frequencies and counts are also split apart and reported for each variant individually (variant #1, #2 etc., depending on the ploidy setting.
- **Overlapping annotations**. This line shows if the SNP is covered by an annotation. The annotation's type and name will displayed. For annotated reference sequences, this information can be used to tell if the SNP is found in e.g. a coding or non-coding region of the genome. **Note** that annotations of type `Variation` and `Source` are not reported.
- **Amino acid change**. If the reference sequence of the is annotated with ORF or CDS annotations, the SNP detection will also report whether the SNP is synonymous or non-synonymous. If the SNP variant changes the amino acid in the protein translation, the new amino acid will be reported (see figure 19.88). Note that adjacent SNPs within the same codon are reported as one SNP in order to determine the impact on the protein level (see section 19.11.4)..

The same information is also recorded in the table. An example of a table is shown in figure 19.88.

In addition to the information shown as annotation, the table also includes the name of the mapping (since the table can include SNPs for many mappings, you need to know which one it belongs to). The table can be **Exported** () as a csv file (comma-separated values) and imported into e.g. Excel. Note that the CSV export includes all the information in the table, regardless of filtering and what has been chosen in the **Side Panel**. If you only want to use a subset of the information, simply select and **Copy** () the information. The columns in the SNP and DIP tables have been synchronized to enable merging in a spreadsheet.

Note that if you make a split view of the table and the mapping (see section 3.2.6), you will be able to browse through the SNPs by clicking in the table. This will cause the view to jump to the position of the SNP.

If you wish to investigate the SNPs further, you can use the filter option (see section C). Figure 19.89 show how to make a filter that only shows homozygote SNPs.



NC_010473 con...

Rows: 174 SNP Detection Table Filter:

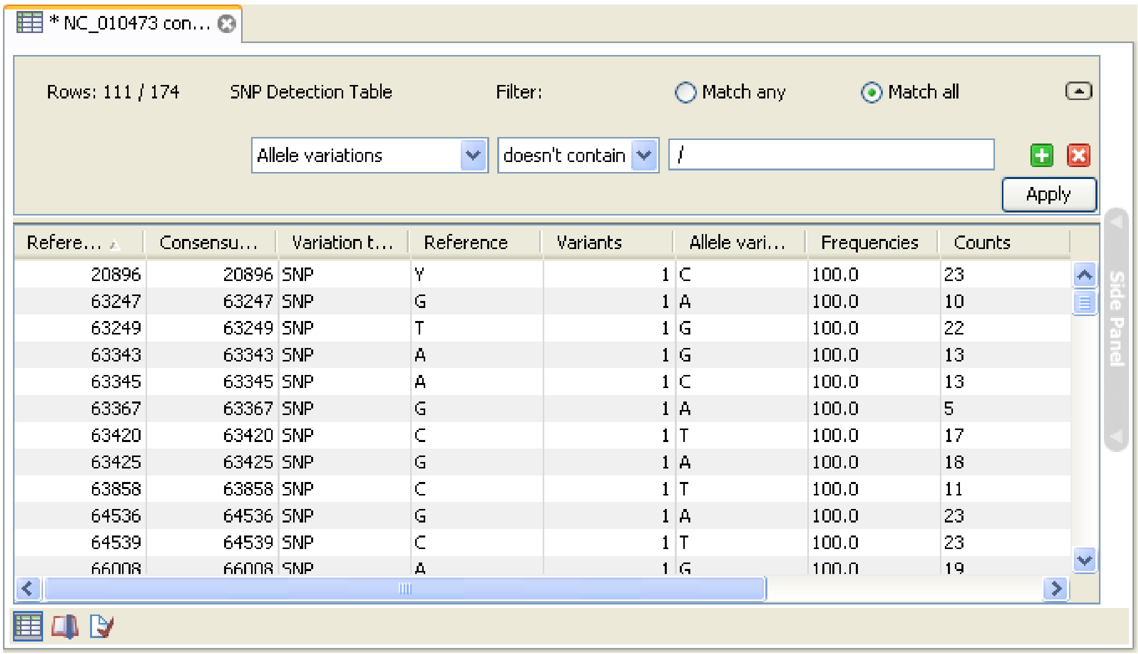
Reference	Allele variations	Frequencies	Counts	Coverage	Overlappin...	Amino acid...
Y	C	100,0	23	23	Gene: rpsT, ...	
G	A	100,0	10	10		
T	G	100,0	22	22		
A	G	100,0	13	13		
A	C	100,0	13	13		
G	A	100,0	5	5		
C	T	100,0	17	17		
G	A	100,0	18	18		
C	T	100,0	11	11	Gene: mraZ, ...	His -> Tyr
G	A	100,0	23	23	Gene: mraW...	
C	T	100,0	23	23	Gene: mraW...	
A	G	100,0	19	19	Gene: ftsI, C...	
T	C	100,0	18	18	Gene: ftsI, C...	
A	G	100,0	18	18	Gene: ftsI, C...	
T	C	100,0	17	17	Gene: ftsI, C...	
T	A	100,0	20	20	Gene: ftsI, C...	
T	C	100,0	24	24	Gene: ftsI, C...	
T	G	100,0	14	14	Gene: ftsI, C...	
G	A	100,0	23	23	Gene: ftsI, C...	
G	A	100,0	25	25	Gene: ftsI, C...	

Table Settings

Column width: Automatic

Show column
- ☐ Reference position
- ☐ Consensus position
- ☐ Variation type
- ☐ Length
- ☑ Reference
- ☐ Variants
- ☑ Allele variations
- ☑ Frequencies
- ☑ Counts
- ☑ Coverage
- ☐ Variant #1
- ☐ Frequency of #1

Figure 19.88: *A table of SNPs.*



* NC_010473 con...

Rows: 111 / 174 SNP Detection Table Filter: ○ Match any ● Match all

Allele variations | doesn't contain | /

Apply

Refere...	Consensu...	Variation t...	Reference	Variants	Allele vari...	Frequencies	Counts
20896	20896	SNP	Y	1	C	100.0	23
63247	63247	SNP	G	1	A	100.0	10
63249	63249	SNP	T	1	G	100.0	22
63343	63343	SNP	A	1	G	100.0	13
63345	63345	SNP	A	1	C	100.0	13
63367	63367	SNP	G	1	A	100.0	5
63420	63420	SNP	C	1	T	100.0	17
63425	63425	SNP	G	1	A	100.0	18
63858	63858	SNP	C	1	T	100.0	11
64536	64536	SNP	G	1	A	100.0	23
64539	64539	SNP	C	1	T	100.0	23
66008	66008	SNP	A	1	G	100.0	19

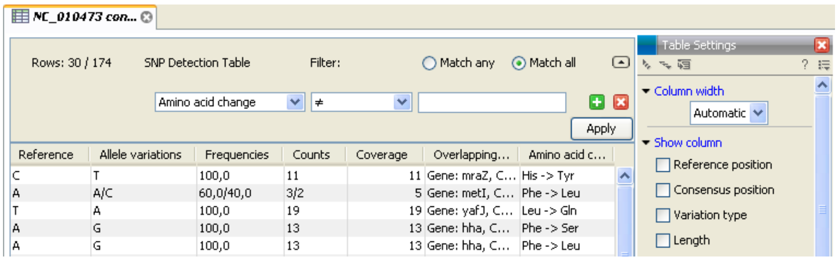
Figure 19.89: *Filtering away the SNPs that have more than one allele variant.*

You can also use the filter to show e.g. nonsynonymous SNPs (filter the **Amino acid change** column to not being empty as shown in figure 19.90).

19.11.4 Adjacent SNPs affecting the same codon

Figure 19.91 shows an example where two adjacent SNPs are found within the same codon.

The *CLC Genomics Workbench* reports these SNPs as one SNP in order to evaluate the effect on



NC_010473 con...

Table Settings

Rows: 30 / 174 SNP Detection Table Filter: ○ Match any ● Match all

Amino acid change | ≠ | Apply

Reference	Allele variations	Frequencies	Counts	Coverage	Overlapping...	Amino acid c...
C	T	100,0	11	11	Gene: mraZ, C...	His -> Tyr
A	A/C	60,0/40,0	3/2	5	Gene: metI, C...	Phe -> Leu
T	A	100,0	19	19	Gene: yafJ, C...	Leu -> Gln
A	G	100,0	13	13	Gene: hha, C...	Phe -> Ser
A	G	100,0	13	13	Gene: hha, C...	Phe -> Leu

Column width: Automatic

Show column

□ Reference position

□ Consensus position

□ Variation type

□ Length

Figure 19.90: *Filtering the SNP table to only display nonsynonymous SNPs.*

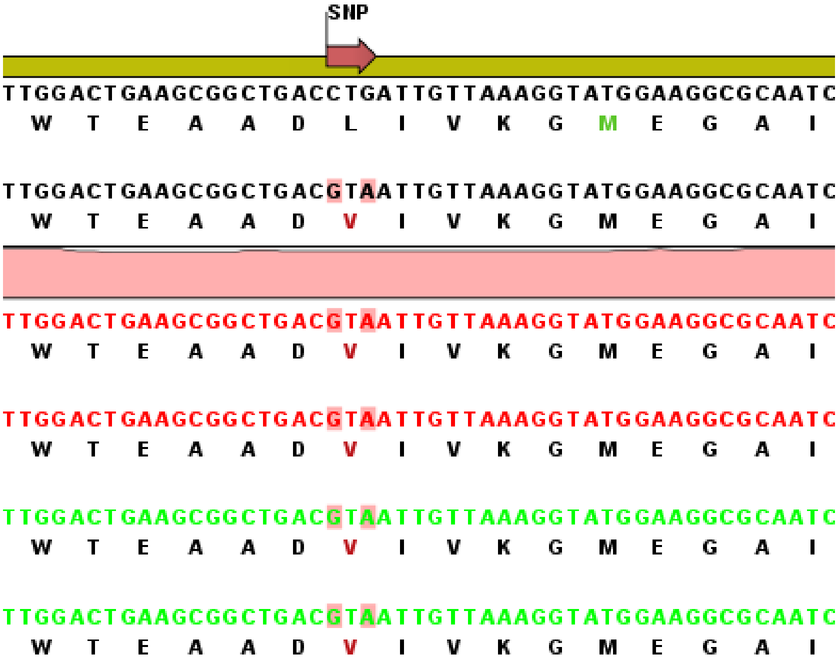

SNP
TTGGACTGAAGCGGCTGACCTGATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D L I V K G M E G A I
TTGGACTGAAGCGGCTGACGTAATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D V I V K G M E G A I
TTGGACTGAAGCGGCTGACGTAATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D V I V K G M E G A I
TTGGACTGAAGCGGCTGACGTAATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D V I V K G M E G A I
TTGGACTGAAGCGGCTGACGTAATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D V I V K G M E G A I
TTGGACTGAAGCGGCTGACGTAATTGTTAAAGGTATGGAAGGCGCAATC
W T E A A D V I V K G M E G A I

Figure 19.91: *Two adjacent SNPs in the same codon.*

the translation to protein. If these SNPs were considered individually, the predicted amino acid change for each individual SNP would not have been correct.

Note that both SNPs need to fulfill the quality filtering and significance criteria to be reported.

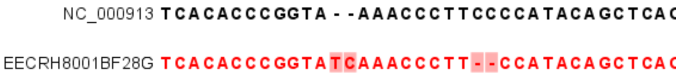
19.12 DIP detection

CLC Genomics Workbench offers automated detection of small deletion/insertion polymorphisms (also known as DIPs) when reads are mapped to a reference.

If you have high coverage in your mapping, you will often find a lot of gaps in the consensus sequence. This is because just a single insertion in one of the reads will cause a gap in all other sequences at this position. The majority of all these gaps should simply be ignored, because they are introduce by single or very few reads because of sequencing errors. The automated DIP detection can be used to find the ones that are significant. If you want to use the consensus sequence for other purposes, you can simply ignore all the gaps (they will disappear once the consensus sequence gets out of the mapping view), and the significant ones can then be annotated as DIPs (see section 19.12.2).

In *CLC Genomics Workbench*, a DIP is a deletion or an insertion of consecutive nucleotides present in experimental sequencing data when compared to a reference sequence. Automated DIP detection is therefore possible only for results from read mapping.

The terms "deletion" and "insertion" are understood as events that have happened to the sequencing sample relative to the reference sequence: when the local alignment between a read and the reference exhibits gaps in the read, nucleotides have been *deleted* (in the read, relative to the reference), and when the local alignment exhibits gaps in the reference sequence, nucleotides have been *inserted* (in the read, relative to the reference). Figure 19.92 shows an insertion (of TC, to the left) and a deletion (of CC, to the right).

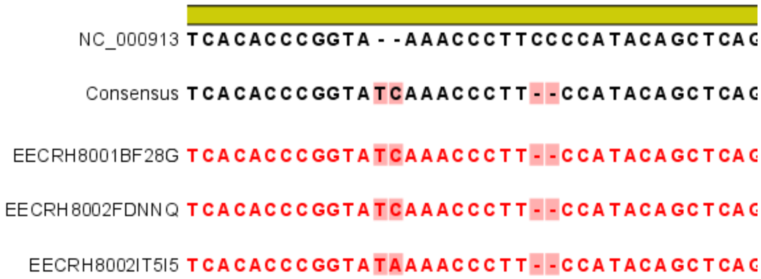


```
NC_000913     TCACACCCGGTA--AAACCCTTCCCCATACAGCTCAC
EECRH8001BF28G TCACACCCGGTATCAAACCCTT--CCATACAGCTCAC
```

Figure 19.92: *Two DIPs, an insertion and a deletion.*

The automated DIP detection in *CLC Genomics Workbench* bases all reported DIPs on DIPs found in *individual* reads. The length of reported deletions and insertions is therefore bounded by the number of insertions and deletions allowed per read by the read mapping algorithm.

In most situations, a DIP in a single read is not sufficient experimental evidence. The *CLC Genomics Workbench* allows you to specify how many reads must cover and agree on a DIP in order for it to be reported by the automated DIP detection. Two reads agree on a deletion if their local alignments to the reference sequence both contain the same number of consecutive gaps aligned to the same reference positions. Likewise, two reads agree on an insertion if their local alignments specify the same number of consecutive gaps at the same position in the reference sequence *and* the nucleotides inserted in the two reads are the same. Figure 19.93 shows some reads disagreeing on an insertion (of TC or TA?, on the left) and agreeing on a deletion (of CC, on the right).



```
NC_000913      TCACACCCGGTA--AAACCCTTCCCCATACAGCTCAG
Consensus      TCACACCCGGTATCAAACCCTT--CCATACAGCTCAG
EECRH8001BF28G TCACACCCGGTATCAAACCCTT--CCATACAGCTCAG
EECRH8002FDNNQ TCACACCCGGTATCAAACCCTT--CCATACAGCTCAG
EECRH8002IT5I5 TCACACCCGGTATAAAACCCTT--CCATACAGCTCAG
```

Figure 19.93: *Disagreement and agreement of DIPs.*

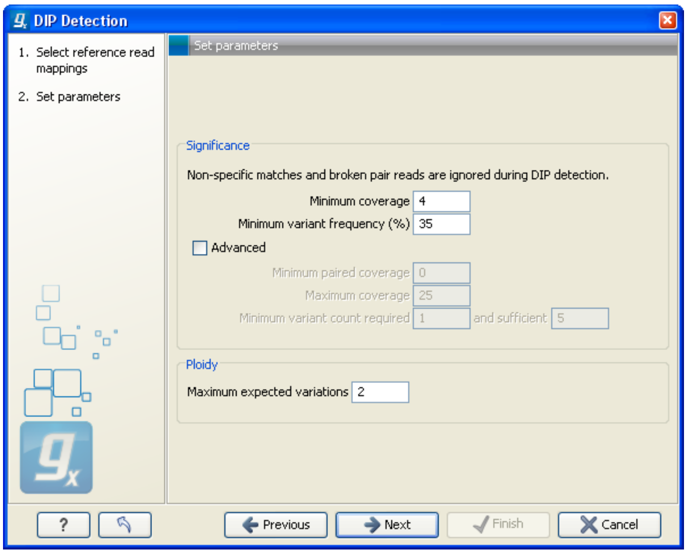
Based on your specifications on what you consider a valid DIP, the DIP detection will scan through the entire mapping and report all the DIPs that meet the requirements:

Toolbox | High-throughput Sequencing ( ) | DIP detection ( )

This opens a dialog where you can select read mapping results ( )/ ( ) to scan for DIPs (see section 19.5 for information on how to map reads to a reference).

19.12.1 Experimental support of a DIP

Clicking **Next** will display the dialog shown in figure 19.94.



Non-specific matches and broken pair reads are ignored during DIP detection.

Minimum coverage 4

Minimum variant frequency (%) 35

□ Advanced

Minimum paired coverage 0

Maximum coverage 25

Minimum variant count required 1 and sufficient 5

Maximum expected variations 2

Figure 19.94: *DIP detection parameters.*

To avoid false positives, the automated DIP detection of *CLC Genomics Workbench* ignores reads that have multiple hit positions on the reference (marked by yeallow color) or reads that come from broken pairs.

In addition, *CLC Genomics Workbench* allows you to specify thresholds for the experimental support of reported DIPs, thus filtering the DIPs found in single, valid reads, see figure 19.94.

- **Minimum coverage**. DIPs called in areas of low coverage will likely result in a higher amount of false positives. Therefore you can set the minimum coverage for a DIP to be called. Note that the coverage is counted as the number of valid reads completely covering the DIP.
- **Minimum variant frequency**. Often reads do not completely agree on a DIP, and you may want to report only the most frequent variants at each DIP site. This threshold can be specified as the percentage of the reads or the absolute number of reads. By default, the frequency in percent is set to 35%, which means that at least 35% of the valid reads covering the DIP site must agree on the DIP for it to be reported. In effect, this means that at most two different variants will be reported at each site, which is reasonable for diploid organisms. If a DIP is frequent enough to be reported, the DIP annotation or table entry will contain information about all other variants which are also frequent enough—even if they are not DIPs.

Below, there is an **Advanced** option letting you specify additional requirements. These will only take effect if the **Advanced** checkbox is checked.

- **Minimum paired coverage**. In based on paired data, more confidence is often attributed to valid paired reads than to single reads. You can therefore set the minimum coverage of valid paired reads in addition to the minimum coverage of all reads. Again, the paired coverage is counted as the number of valid reads completely covering the DIP (the space between mating pairs does not cover anything.) Note that regardless of this setting, reads from broken pairs are never considered for DIP detection.
- **Maximum coverage**. Read coverage often displays peaks in repetitive regions where the alignment is not very trustworthy. Setting the maximum coverage threshold a little higher

than the expected average coverage (allowing for some variation) can be helpful in ruling out false positives from such regions.

- **Minimum variant counts**. This option is the threshold for the number of reads that display a DIP at a given position. In addition to the percentage setting in the simple panel above, these settings are based on absolute counts. If the count required is set to 3, and the sufficient count is set to 5, it means that **even though** less than the required percentage of the reads have a DIP, it will still be reported as a DIP if at least 5 reads have it. However, if the count is 2, the DIP will not be called, regardless the percentage setting. This distinction is especially useful with deep sequencing data where you have very high coverage and many different alleles. In this case, the percentage threshold is not suitable for finding valid DIPs in a small subset of the data. If you are not interested in reporting DIPs based on counts but only rely on the relative frequency, you can simply set the sufficient count number very high.

- **Maximum expected variations**. This is not a filtering option, but is related to the minimum variant frequency setting. By setting the frequency threshold low enough to allow more variants than the ploidy of the organism sequenced, you can use the maximum expected variations setting to mark reported DIPs as "complex", if they involve more variations then expected from the ploidy. For example, using a minimum variant frequency of 30% with a diploid organism, you are allowing DIPs with up to 3 variations, and then by setting the maximum expected variations count to 2 (the default), any DIPs with 3 variations will be marked as complex (see below).

19.12.2 Reporting the DIPs

When you click **Next**, you will be able to specify how the DIPs should be reported:

- **Annotate reference sequence(s)**. This will add an annotation for each DIP to the reference sequences in the input.

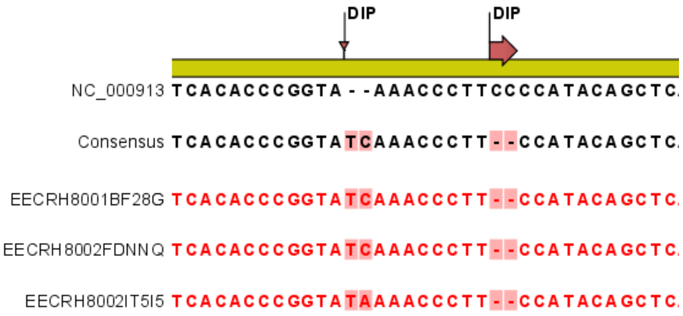
- **Annotate consensus sequence(s)**. This will add an annotation for each DIP to the consensus sequences in the input. In either way, DIP annotations contain the following information:

  - **Reference position**. The first position of the DIP in the reference sequence.
  - **Consensus position**. The first position of the DIP in the consensus sequence.
  - **Variation type**. Will be "DIP" or "Complex DIP", depending on the value of the maximum expected variations setting and the actual number of variations found at the DIP site.
  - **Length**. The length of the DIP. Note that only small deletions and insertions are found. This is because the DIP detection is based on the alignment of the reads generated by the mapping process, and the mapping only allows a few insertions/deletions (see section 19.5 for information on how to map reads to a reference).
  - **Reference**. The residues found in the reference sequence (either gaps for insertions or bases for deletions).
  - **Variants**. The number of variants among the reads.

  - **Allele variation**. The variations found in the reads at the DIP site. Contains only those variations whose frequency is at least that specified by the minimum variant frequency setting.
  - **Frequencies**. The frequencies of the variations, both absolute (counts) and relative (percentage of coverage).
  - **Coverage**. The number of valid reads completely covering the DIP site.
  - **Variant numbers and frequencies**. The information from the Allele variations, frequencies and counts are also split apart and reported for each variant individually (variant #1, #2 etc., depending on the ploidy setting.
  - **Overlapping annotations**. Says if the DIP is covered, in part or in whole, by an annotation. The annotation's type and name will displayed. For annotated reference sequences, this information can be used to tell if the DIP is found in e.g. a coding or non-coding region of the genome. **Note** that annotations of type `Variation` and `Source` are not reported.
  - **Amino acid change**. If the reference sequence of is annotated with ORF or CDS annotations, the DIP detection will also report whether the DIP changes the amino acid sequence resulting from translation, and, if so, whether the change involves frame-shifting.

- **Create table**. This will create a table showing all the DIPs found. The table will provide a valuable overview, whereas the annotations are useful for detailed inspection of a DIP, and also if the annotated sequences are used for further analysis in the *CLC Genomics Workbench*.

Figure 19.95 shows the result of a DIP detection output as annotations on the reference sequence. The DIP detection found the DIPs of figure 19.93.



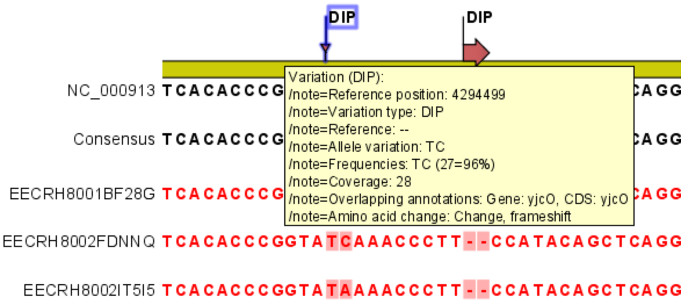
```
     NC_000913 TCACACCCGGTA--AAACCCTTCCCCATACAGCTC
     Consensus TCACACCCGGTATCAAACCCTT--CCATACAGCTC
EECRH8001BF28G TCACACCCGGTATCAAACCCTT--CCATACAGCTC
EECRH8002FDNNQ TCACACCCGGTATCAAACCCTT--CCATACAGCTC
EECRH8002IT5I5 TCACACCCGGTATAAAACCCTT--CCATACAGCTC
```

Figure 19.95: *DIPs detected witin a coding region.*

The DIPs occur within a coding region (identified by the long yellow annotation) and you can see that they both shift the frame of the translation, since their sizes are not divisible by 3. Placing your mouse on the annotations will reveal detailed information about the DIPs as shown in figure 19.96.

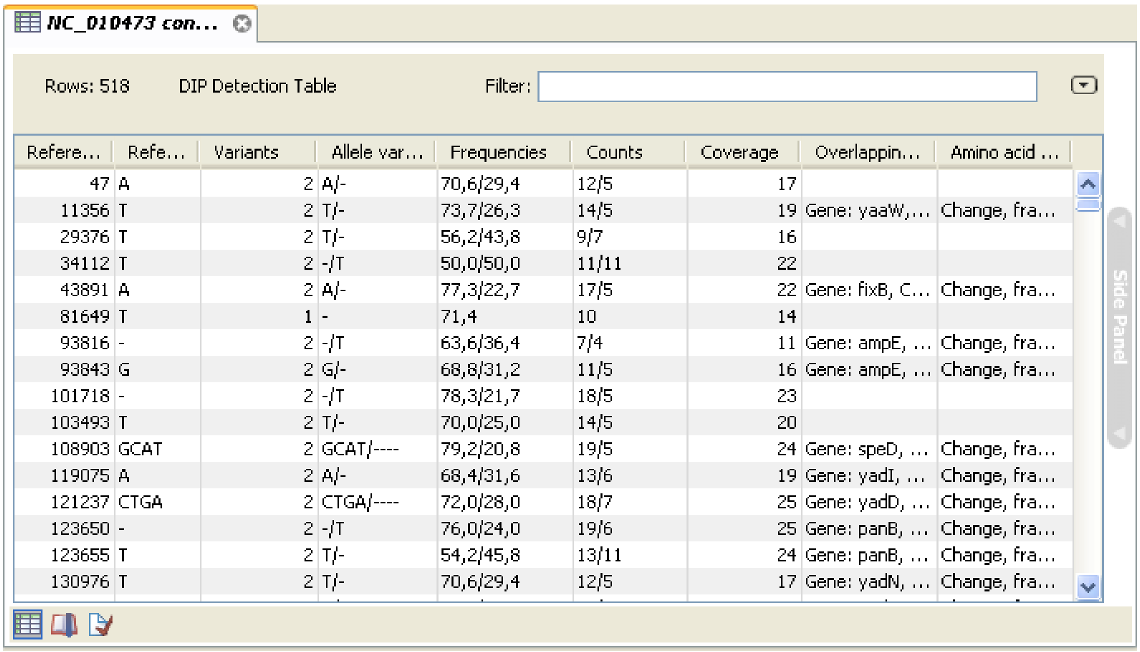
The same information is also recorded in the table output. An example of a table is shown in figure 19.97.

In addition to the information shown as annotation, the table also includes the name of the mapping (since the table can include DIPs for many references, you need to know which one it


DIP
DIP
NC_000913 TCACACCCG
CAGG
Consensus TCACACCCG
CAGG
EECRH8001BF28G TCACACCCG
CAGG
EECRH8002FDNNQ TCACACCCGGTATCAAACCCTT--CCATACAGCTCAGG
EECRH8002IT5I5 TCACACCCGGTATAAAACCCTT--CCATACAGCTCAGG

Variation (DIP):
/note=Reference position: 4294499
/note=Variation type: DIP
/note=Reference: --
/note=Allele variation: TC
/note=Frequencies: TC (27=96%)
/note=Coverage: 28
/note=Overlapping annotations: Gene: yjcO, CDS: yjcO
/note=Amino acid change: Change, frameshift

Figure 19.96: *A DIP annotation with detailed information.*



Rows: 518 DIP Detection Table Filter:

Refere...	Refe...	Variants	Allele var...	Frequencies	Counts	Coverage	Overlappin...	Amino acid ...
47	A	2	A/-	70,6/29,4	12/5	17		
11356	T	2	T/-	73,7/26,3	14/5	19	Gene: yaaW,...	Change, fra...
29376	T	2	T/-	56,2/43,8	9/7	16		
34112	T	2	-/T	50,0/50,0	11/11	22		
43891	A	2	A/-	77,3/22,7	17/5	22	Gene: fixB, C...	Change, fra...
81649	T	1	-	71,4	10	14		
93816	-	2	-/T	63,6/36,4	7/4	11	Gene: ampE, ...	Change, fra...
93843	G	2	G/-	68,8/31,2	11/5	16	Gene: ampE, ...	Change, fra...
101718	-	2	-/T	78,3/21,7	18/5	23		
103493	T	2	T/-	70,0/25,0	14/5	20		
108903	GCAT	2	GCAT/----	79,2/20,8	19/5	24	Gene: speD, ...	Change, fra...
119075	A	2	A/-	68,4/31,6	13/6	19	Gene: yadI, ...	Change, fra...
121237	CTGA	2	CTGA/----	72,0/28,0	18/7	25	Gene: yadD, ...	Change, fra...
123650	-	2	-/T	76,0/24,0	19/6	25	Gene: panB, ...	Change, fra...
123655	T	2	T/-	54,2/45,8	13/11	24	Gene: panB, ...	Change, fra...
130976	T	2	T/-	70,6/29,4	12/5	17	Gene: yadN, ...	Change, fra...

Figure 19.97: *A table of DIPs.*

belongs to). The table can be **Exported** (📤) as a csv file (comma-separated values) and imported into e.g. Excel. Note that the CSV export includes all the information in the table, regardless of filtering and what has been chosen in the **Side Panel**. If you only want to use a subset of the information, simply select and **Copy** (📋) the information. The columns in the SNP and DIP tables have been synchronized to enable merging in a spreadsheet.

Note that if you make a split view of the table and the mapping (see section 3.2.6), you will be able to browse through the DIPs by clicking in the table. This will cause the view to jump to the position of the DIP.

19.13 ChIP sequencing

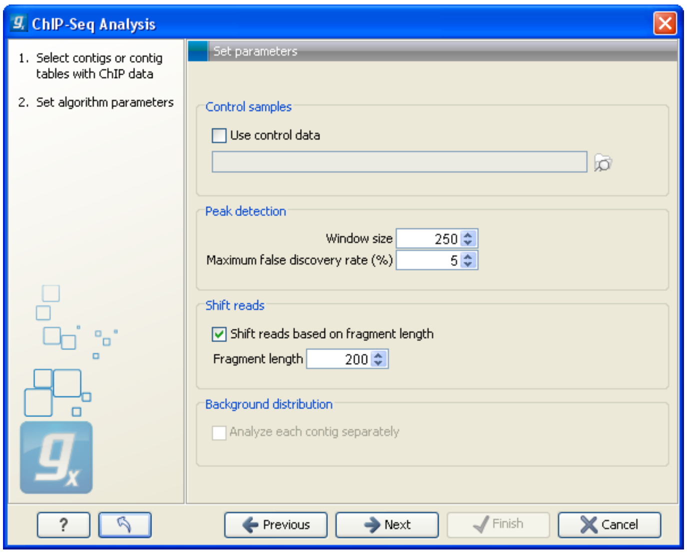
*CLC Genomics Workbench*can perform analysis of Chromatin immunoprecipitation sequencing (ChIP-Seq) data based on the information contained in a single sample subjected to immunoprecipitation (ChIP-sample) or by comparing a ChIP-sample to a control sample where the immunoprecipitation step is omitted. The first step in a ChIP-Seq analysis is to map the reads to a reference (see section 19.5) which maps your reads against one or more specified reference sequences. If both a ChIP- and a control sample is used, these must be mapped separately to produce separate ChIP- and control samples. These samples are then used as input to the ChIP-Seq tool which surveys the pattern in coverage to detect significant peaks:

> **Toolbox** | **High-throughput Sequencing ()** | **ChIP-Seq Analysis ()**

This opens a dialog where you can select one or more mapping results ()/ () to use as ChIP-samples. Control samples are selected in the next step.

19.13.1 Peak finding and false discovery rates

Clicking **Next** will display the dialog shown in figure 19.98.


ChIP-Seq Analysis

1. Select contigs or contig tables with ChIP data
2. Set algorithm parameters

Set parameters

Control samples

☐ Use control data

Peak detection

Window size 250

Maximum false discovery rate (%) 5

Shift reads

☑ Shift reads based on fragment length

Fragment length 200

Background distribution

☐ Analyze each contig separately

Figure 19.98: *Peak finding and false discovery rates.*

If the option to include control samples is included, the user must select the appropriate sample to use as control data. If the mapping is based on several reference sequences, the Workbench will automatically match the ChIP-samples and controls based on the length of the reference sequences.

The peak finding algorithm includes the following steps:

- Calculate the null distribution of background sequencing signal
- Scan the mappings to identify candidate peaks with a higher read count than expected from the null distribution
- Merge overlapping candidate peaks
- Refine the set of candidate peaks based on the count and the spatial distribution of reads of forward and reverse orientation within the peaks

The estimation of the null distribution of coverage and the calculation of the false discovery rates are based on the **Window size** and **Maximum false discovery rate (%)** parameters. The **Window size** specifies the width of the window that is used to count reads both when the null distribution is estimated and for the subsequent scanning for candidate peaks.

The **Maximum false discovery rate** specifies the maximum proportion of false positive peaks that you are willing to accept among your called peaks. A value of 10 % means that you are willing to accept that 10 % of the peaks called are expected to be false discoveries.

To estimate the false discovery rate (FDR) we use the method of [Ji et al., 2008] (see also Supplementary materials of the paper).

In the case where only a ChIP-sample is used, a negative binomial distribution is fitted to the counts from low coverage regions. This distribution is used as a null distribution to obtain the numbers of windows with a particular count of reads that you would expect in the absence of significant binding. By comparing the number of windows with a specific count you expect to see under the null distribution and the number you actually see in your data, you can calculate a false discovery rate for a given read count for a given window size as: 'fraction of windows with read count expected under the null distribution'/'fraction of windows with read count observed'.

In the case where both a ChIP- and a control sample are used, a sampling ratio between the samples is first estimated, using only windows in which the total numbers of reads (that is, the sum of those in the sample and those in the control) is small. The sampling ratio is estimated as the ratio of the cumulated sample read counts ($c^{sample} = \sum_i k_i^{sample}$) to cumulated control read counts ($c^{control} = \sum_i k_i^{control}$) in these windows. The sampling ratio is used to estimate the proportion of the reads that are expected to be ChIP-sample reads under the null distribution, as $p_0 = c^{sample}/(c^{sample} + c^{control})$. For a given total read count, *n*, of a window, the numbers of reads expected in the ChIP-sample under the null distribution can then be estimated from the binomial distribution with parameters *n* and p_0. By comparing the expected and observed numbers, a false discovery rate can then be calculated. Note, that when a control sample is used different null-distributions are estimated for different total read counts, *n*.

In both cases, the user can specify whether the null distribution should be estimated separately for each reference sequence by checking the option **Analyze each reference separately**.

Because the ChIP-seq experimental protocol selects for sequencing input fragments that are centered around a DNA-protein binding site it is expected that true peaks will exhibit a signature distribution where forward reads are found upstream of the binding site and reverse reads are found downstream of the binding site leading to reduced coverage at the exact binding site. For this reason, the algorithm allows you to shift forward reads towards the 3' end and reverse reads towards the 5' end in order to generate a more marked peak prior to the peak detection step. This is done by checking the **Shift reads based on fragment length** box. To shift the reads you also need to input the expected length of the sequencing input fragments by setting the **Fragment length** parameter, this is the size of the fragment isolated from gel (L in the illustration below).

The illustration below shows a peak where the forward reads are in one window and the reverse reads fall in another window (window 1 and 3).

```
---------------------------------------------------------- reference
        -----------------------|-------------------        (actual sequenced fragment length = L bp)
        ---->                                              reads
         ---->                                             reads
                                              <---         reads
                                            <---           reads.

|---------------------|--------------------|-------------  window size W
         1                       2                  3
```

If the reads are not shifted, the algorithm will count 2 reads in window 1 and 3. But if the forward reads are shifted 0.5XL to the right and reverse reads are shifted 0.5xL to left, the algorithm will find 4 reads in window 2 as shown below:

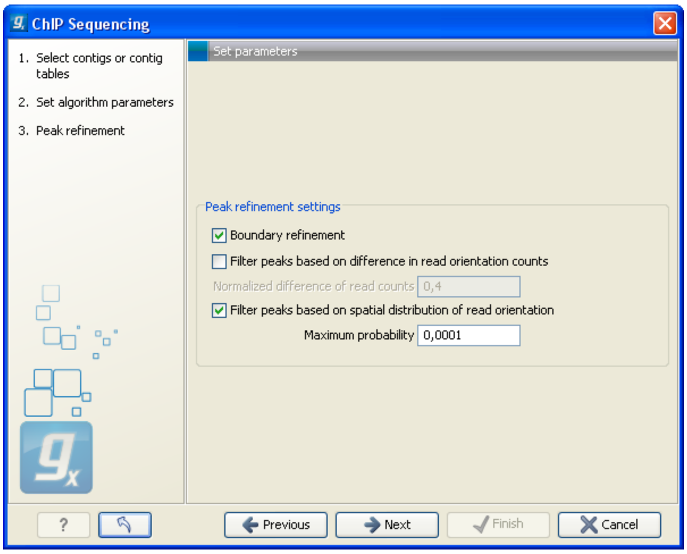
```
--------------------------------------------------------- reference
        ----------------------|------------------         (actual sequenced fragment length = L basepairs)
                              ---->                         reads
                               ---->                        reads
                           <---                             reads
                          <---                              reads

|--------------------|--------------------|-------------  window size W
         1                      2                  3
```

So shifting reads will increase the signal to noise ratio.

The following peak refinement step, the reporting of the peak and the visualization will use the original position of the reads, so the shifting is only a virtual shift performed as part of the peak detection.

19.13.2 Peak refinement

Clicking **Next** will display the dialog shown in figure 19.99.



ChIP Sequencing

1. Select contigs or contig tables
2. Set algorithm parameters
3. Peak refinement

Set parameters

Peak refinement settings

☑ Boundary refinement

□ Filter peaks based on difference in read orientation counts

Normalized difference of read counts 0,4

☑ Filter peaks based on spatial distribution of read orientation

Maximum probability 0,0001

Figure 19.99: *Peak refinement settings.*

This dialog presents the parameters and options that can be used to refine the set of candidate peaks discovered when scanning the read mapping. All three refinement options again utilize the fact that coverage around a true DNA-protein binding site is expected to exhibit a signature distribution where forward reads are found upstream of the binding site and reverse reads are found downstream of the binding site. Peak refinement can be performed both with- and without a control sample but the algorithm only uses information contained in the reads from the ChIP-samples, not the control samples.

If the **Boundary refinement** option is checked, the algorithm will estimate the position of the DNA-protein binding interaction and place the resulting annotations on this region, rather than on the region where a peak in coverage is found. A center of sequencing intensity is defined for all forward reads as the median value of the center points of all forward reads and likewise for all reverse reads. The "refined peak" is thus defined as the region between these two points.

One of the advantages of including this boundary refinement is that shorter regions can be given as input to subsequent pattern discovery analysis.

By checking the **Filter peaks based on difference in read orientation counts** the algorithm will calculate the normalized difference in the number of forward and reverse reads within a peak as

$$\frac{|\text{count forward reads - count reverse reads}|}{\text{count forward reads + count reverse reads}}$$

The desired maximum value of this parameter can be set in the **Normalized difference of read counts** field and any candidate peak with a value above this will then be dismissed. Setting a low value will ensure that peaks are only called if there is a well balanced number of forward and reverse reads.

As an example if you have 15 forward reads and 5 reverse reads, you will end up with a value of 0.5. With the default limit set to 0.4, a peak like that would be excluded.

By checking the **Filter peaks based on spatial distribution of read orientation** the algorithm will evaluate how clearly separated the location of forward and reverse reads are within a peak. This is done via the Wilcoxon rank-sum test (see http://en.wikipedia.org/wiki/Mann-Whitney-Wilcoxon_test). The null hypothesis here is that the positions of forward and reverse reads within a peak are drawn from the same distribution i.e. that their locations are not significantly different and the alternative hypothesis is that the forward reads have a sum of ranked positions that is shifted to lower positions than the reverse reads. Peaks will be dismissed if the probability of the null hypothesis exceeds the value set in the **Maximum probability** field.

Setting a low **Maximum probability** will ensure that peaks are only called if there is a clear signature distribution where forward reads are found upstream of reverse reads within the peak.

A general comment about peak filtering is that the relevant statistics are all reported in the peak table that the algorithm outputs. If it is desirable to explore a large set of candidate peaks it is recommended to use no or relatively loose filtering criteria and then use the advanced table filtering options to explore the effect of the different parameters (see section C). It may be desirable to omit the addition of annotations in this exploratory analysis and rely on the information in the table instead. Once a desired set of parameters is found, the algorithm can be rerun using these as filtering criteria to add annotations to the reference sequence and to produce a final list of peaks.

19.13.3 Reporting the results

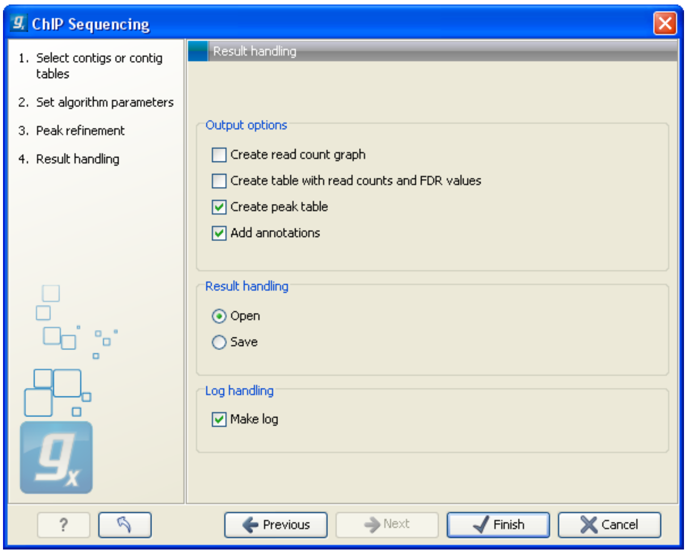
When you click **Next**, you will be able to specify how the results should be reported (see figure 19.100).

The different output options are described in detail below. Note, that it is not possible to output a graph and table of read counts in the case where a control sample is used. These options are therefore disabled in this case.

Graph and table of background distribution and false discovery statistics

An example of a FDR graph based on a single ChIP-sample is shown in figure 19.101.

The graph shows the estimated background distribution of read counts in discrete windows and the observed counts and can thus be used to inspect how well the estimated distribution fits the observed pattern of coverage.



Output options

- ☐ Create read count graph
- ☐ Create table with read counts and FDR values
- ☑ Create peak table
- ☑ Add annotations

Result handling

- ● Open
- ○ Save

Log handling

- ☑ Make log

Figure 19.100: *Output options.*

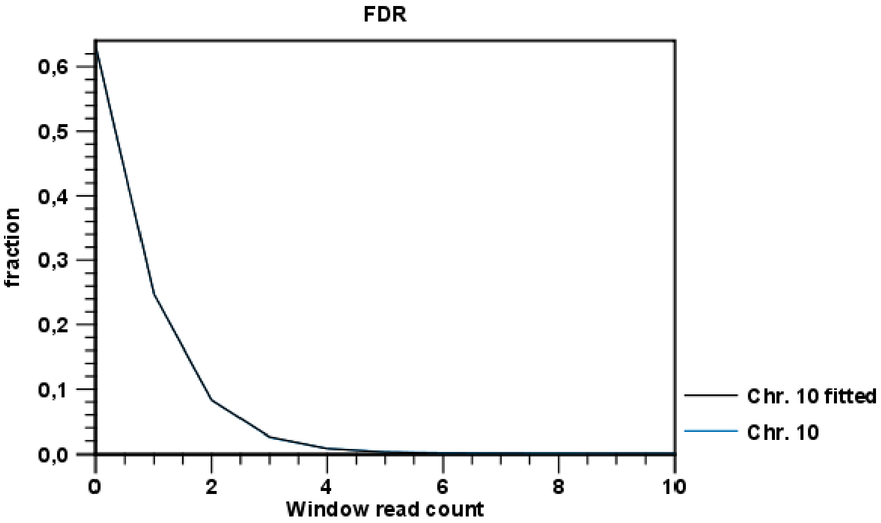


Figure 19.101: *FDR graph.*

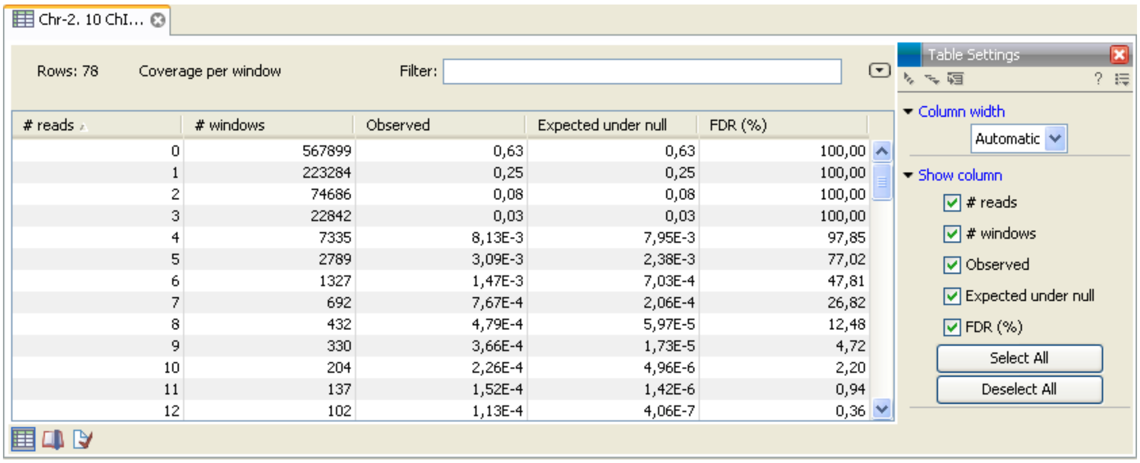
The FDR table displays the observed and expected fraction of windows with a given read count and also shows the rate of false discovery related to a given level of coverage within a window:

- **# reads** - the number of reads within a window.
- **# windows** - the number of windows with the given read count. A window of a fixed width is slid across the sequence. For every window position the number of reads in that window is recorded and stored as the read count. After this, the windows are counted based on their recorded read counts. # windows of read count x is thus the number of windows that were found to contain x reads during this process. This is done to establish the background distribution of coverage and to evaluate the fit of the estimated distribution.
- **Observed** - the observed faction of windows with the given read count.
- **Expected under null** - the expected fraction of windows with a given read count, under the

null distribution.

- **FDR %** - the false discovery rate which is the fraction of the peaks with the given read count that can be expected to be false positives.

An example is shown in figure 19.102.



Rows: 78 Coverage per window Filter:

☑ # reads
☑ # windows
☑ Observed
☑ Expected under null
☑ FDR (%)

# reads	# windows	Observed	Expected under null	FDR (%)
0	567899	0,63	0,63	100,00
1	223284	0,25	0,25	100,00
2	74686	0,08	0,08	100,00
3	22842	0,03	0,03	100,00
4	7335	8,13E-3	7,95E-3	97,85
5	2789	3,09E-3	2,38E-3	77,02
6	1327	1,47E-3	7,03E-4	47,81
7	692	7,67E-4	2,06E-4	26,82
8	432	4,79E-4	5,97E-5	12,48
9	330	3,66E-4	1,73E-5	4,72
10	204	2,26E-4	4,96E-6	2,20
11	137	1,52E-4	1,42E-6	0,94
12	102	1,13E-4	4,06E-7	0,36

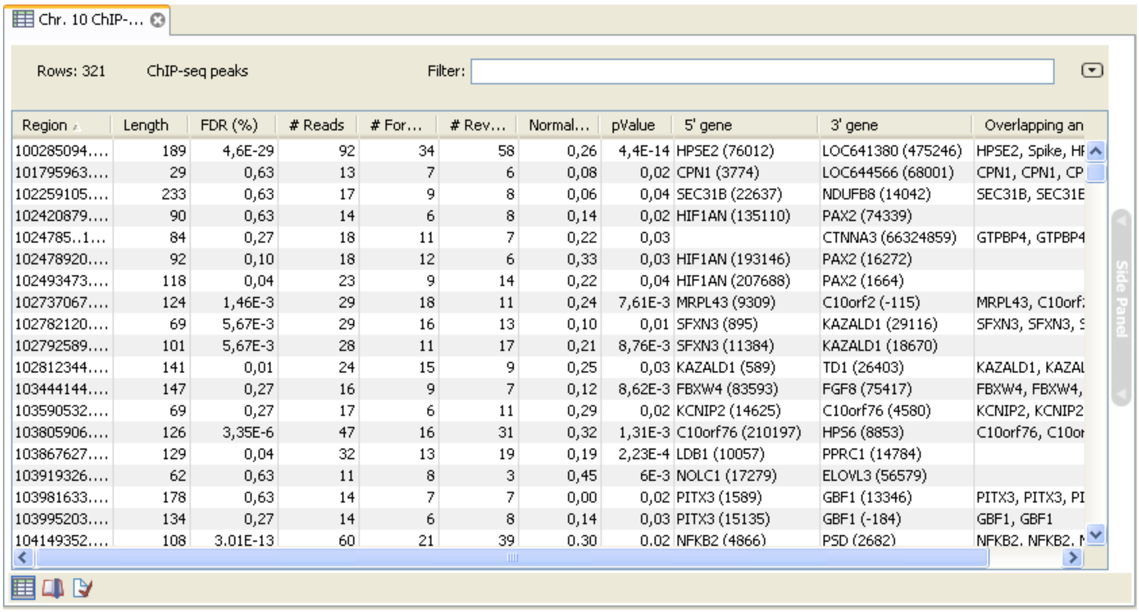
Figure 19.102: *FDR table.*

From this table you can see that less than 5% of the called peaks with 9 reads can be expected to be false discoveries and for peaks with 11 reads the FDR is less than 1%.

Peak table and annotations

The main result is the table showing the peaks and the annotations added to the reference sequence.

An example of a peak table is shown in figure 19.103.



Rows: 321 ChIP-seq peaks Filter:

Region	Length	FDR (%)	# Reads	# For...	# Rev...	Normal...	pValue	5' gene	3' gene	Overlapping an
100285094....	189	4,6E-29	92	34	58	0,26	4,4E-14	HPSE2 (76012)	LOC641380 (475246)	HPSE2, Spike, HF
101795963....	29	0,63	13	7	6	0,08	0,02	CPN1 (3774)	LOC644566 (68001)	CPN1, CPN1, CP
102259105....	233	0,63	17	9	8	0,06	0,04	SEC31B (22637)	NDUFB8 (14042)	SEC31B, SEC31E
102420879....	90	0,63	14	6	8	0,14	0,02	HIF1AN (135110)	PAX2 (74339)	
1024785..1...	84	0,27	18	11	7	0,22	0,03		CTNNA3 (66324859)	GTPBP4, GTPBP4
102478920....	92	0,10	18	12	6	0,33	0,03	HIF1AN (193146)	PAX2 (16272)	
102493473....	118	0,04	23	9	14	0,22	0,04	HIF1AN (207688)	PAX2 (1664)	
102737067....	124	1,46E-3	29	18	11	0,24	7,61E-3	MRPL43 (9309)	C10orf2 (-115)	MRPL43, C10orf:
102782120....	69	5,67E-3	29	16	13	0,10	0,01	SFXN3 (895)	KAZALD1 (29116)	SFXN3, SFXN3, S
102792589....	101	5,67E-3	28	11	17	0,21	8,76E-3	SFXN3 (11384)	KAZALD1 (18670)	
102812344....	141	0,01	24	15	9	0,25	0,03	KAZALD1 (589)	TD1 (26403)	KAZALD1, KAZAL
103444144....	147	0,27	16	9	7	0,12	8,62E-3	FBXW4 (83593)	FGF8 (75417)	FBXW4, FBXW4,
103590532....	69	0,27	17	6	11	0,29	0,02	KCNIP2 (14625)	C10orf76 (4580)	KCNIP2, KCNIP2
103805906....	126	3,35E-6	47	16	31	0,32	1,31E-3	C10orf76 (210197)	HPS6 (8853)	C10orf76, C10or
103867627....	129	0,04	32	13	19	0,19	2,23E-4	LDB1 (10057)	PPRC1 (14784)	
103919326....	62	0,63	11	8	3	0,45	6E-3	NOLC1 (17279)	ELOVL3 (56579)	
103981633....	178	0,63	14	7	7	0,00	0,02	PITX3 (1589)	GBF1 (13346)	PITX3, PITX3, PI
103995203....	134	0,27	14	6	8	0,14	0,03	PITX3 (15135)	GBF1 (-184)	GBF1, GBF1
104149352....	108	3.01E-13	60	21	39	0.30	0.02	NFKB2 (4866)	PSD (2682)	NFKB2. NFKB2. N

Figure 19.103: *ChIP sequencing peak table.*

The table includes information about each peak that has been found:

- **Name**. If the mapping was based on more than one reference sequence, the name of the reference sequence in question will be shown here.
- **Region**. The position of the peak. To find that position in the ChIP-sample mapping, you can make a split view of the table and the mapping (see section 3.2.6). You will then be able to browse through the peaks by clicking in the table. This will cause the view to jump to the position of the peak region.
- **Length**. The length of the peak.
- **FDR (%)**. The false discovery rate for the peak (learn more in section 19.13.1).
- **# Reads**. The total number of reads covering the peak region.
- **# Forward reads**. The number of forward reads covering the peak region.
- **# Reverse reads**. The number of reverse reads covering the peak region. The normalized difference in the count of forward-reverse reads is calculated based on these numbers (see figure 19.99).
- **Normalized difference**. See section 19.13.2.
- **P-value**. The p-value is for the Wilcoxon rank sum test for the equality of location of forward and reverse reads in a peak. See section 19.13.2.
- **Max forward coverage**. The refined region described in section 19.13.2 is calculated based on the maximum coverage of forward and reverse reads.
- **Max reverse coverage**. See previous.
- **Refined region**. The refined region.
- **Refined region length**. The length of the refined region.
- **5' gene**. The nearest gene upstream, based on the start position of the gene. The number in brackets is the distance from the peak to the gene start position.
- **3' gene**. The nearest gene downstream, based on the start position of the gene. The number in brackets is the distance from the peak to the gene start position.
- **Overlapping annotations**. Displays any annotations present on the reference sequence that overlap the peak.

Note that if you make a split view of the table and the mapping (see section 3.2.6), you will be able to browse through the peaks by clicking in the table. This will cause the view to jump to the position of the peak.

An example of a peak is shown in figure 19.104.

If you want to extract the sequence of all the peak regions to a list, you can use the **Extract Annotations** plug-in (see `http://www.clcbio.com/index.php?id=938`) to extract all annotations of the type "Binding site".

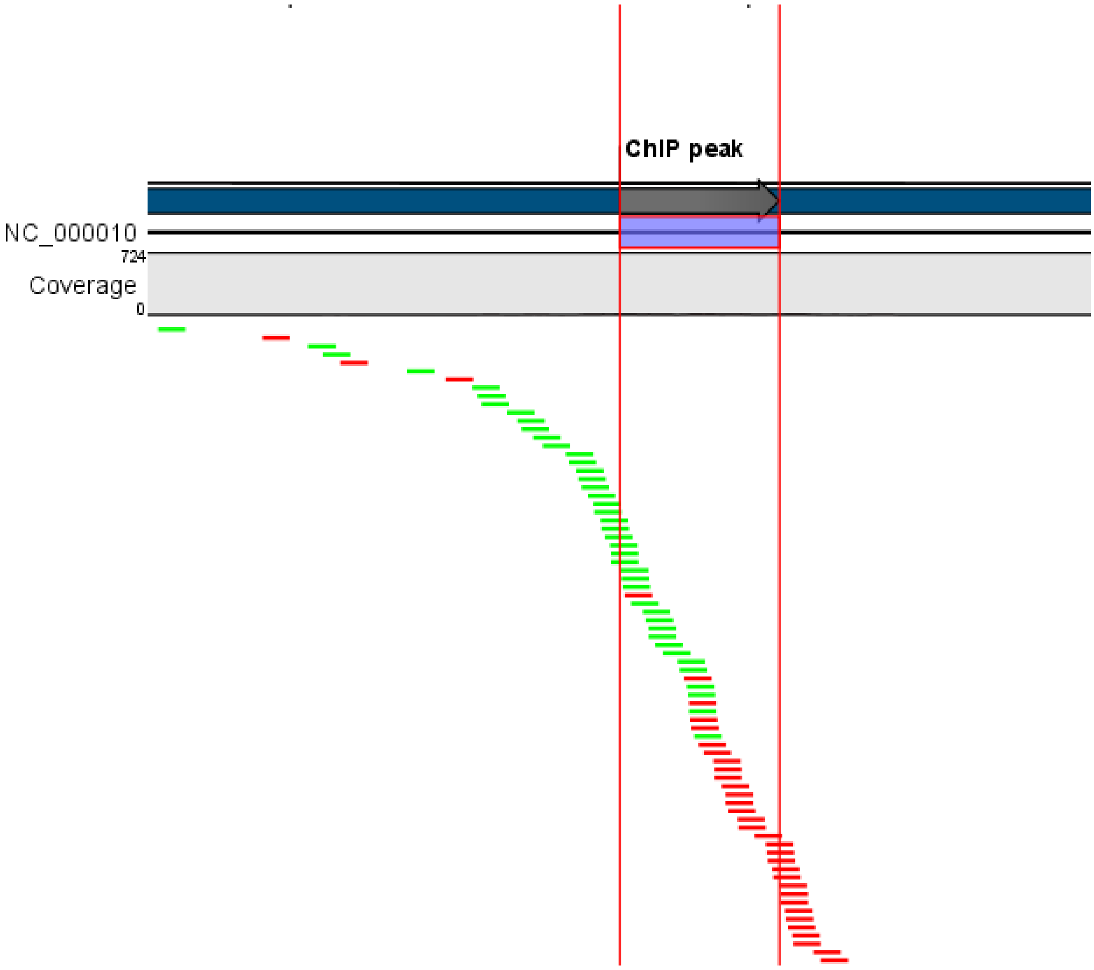


Figure 19.104: *Inspecting an annotated peak. The green lines represent forward reads and the red lines represent reverse reads.*

19.14 RNA-Seq analysis

Based on an annotated reference genome and mRNA sequencing reads, the *CLC Genomics Workbench* is able to calculate gene expression levels as well as discover novel exons.

The approach taken by the *CLC Genomics Workbench* is based on [Mortazavi et al., 2008].

The RNA-Seq analysis is done in several steps: First, all genes are extracted from the reference genome (using annotations of type *gene*). Other annotations on the gene sequences are preserved (e.g. CDS information about coding sequences etc). Next, all annotated transcripts are extracted. If there are several annotated splice variants, they are all extracted. Note that the *mRNA* annotation type is used for extracting the exon-exon boundaries, so your reference sequence needs to be annotated with this kind of annotations (for eukaryotes only - prokaryotes only need the gene annotations).

An example is shown in figure 19.105.

This is a simple gene with three exons and two splice variants. The transcripts are extracted as shown in figure 19.106.

Next, the reads are mapped against all the transcripts plus the entire gene (see figure 19.107).

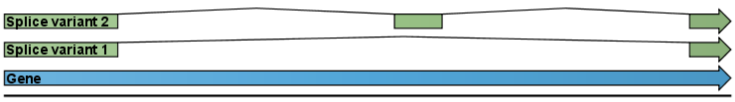


Figure 19.105: *A simple gene with three exons and two splice variants.*

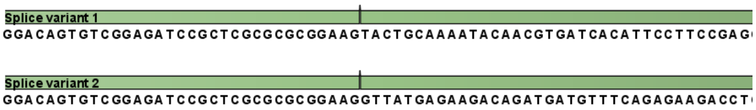

Splice variant 1
GGACAGTGTCGGAGATCCGCTCGCGCGCGGAAGTACTGCAAAATACAACGTGATCACATTCCTTCCGAG
Splice variant 2
GGACAGTGTCGGAGATCCGCTCGCGCGCGGAAGGTTATGAGAAGACAGATGATGTTTCAGAGAAGACCT

Figure 19.106: *All the exon-exon junctions are joined in the extracted transcript.*

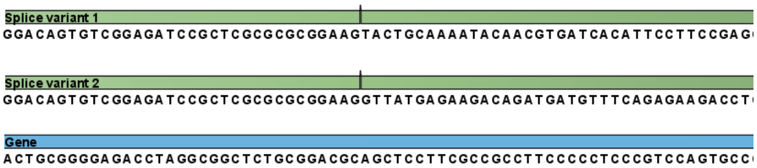

Splice variant 1
GGACAGTGTCGGAGATCCGCTCGCGCGCGGAAGTACTGCAAAATACAACGTGATCACATTCCTTCCGAG
Splice variant 2
GGACAGTGTCGGAGATCCGCTCGCGCGCGGAAGGTTATGAGAAGACAGATGATGTTTCAGAGAAGACCT
Gene
ACTGCGGGGAGACCTAGGCGGCTCTGCGGACGCAGCTCCTTCGCCGCCTTCCCCCTCCCGTCCAGTGCC

Figure 19.107: *The reference for mapping: all the exon-exon junctions and the gene.*

From this mapping, the reads are categorized and assigned to the genes (elaborated later in this section), and expression values for each gene and each transcript are calculated. After that, putative exons are identified.

Details on the process are elaborated below when describing the user interface. To start the RNA-Seq analysis analysis:

Toolbox | High-throughput Sequencing () | RNA-Seq Analysis ()

This opens a dialog where you select the sequencing reads (not the reference genome or transcriptome). The sequencing data should be imported as described in section 19.1.

If you have several different samples that you wish to measure independently and compare afterwards, you should run the analysis in batch mode (see section 9.1).

Click **Next** when the sequencing data is listed in the right-hand side of the dialog.

19.14.1 Defining reference genome and mapping settings

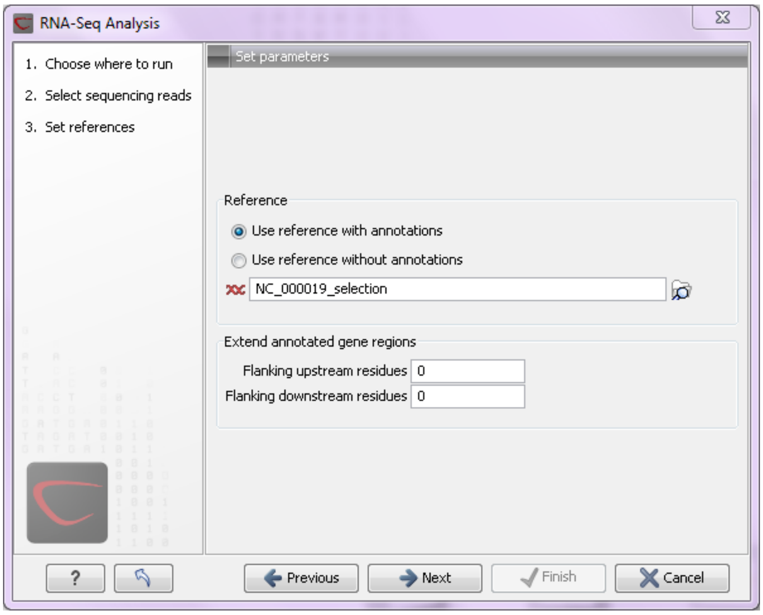
You are now presented with the dialog shown in figure 19.108.

At the top, there are two options concerning how the reference sequences are annotated:

- **Use reference with annotations**. Choosing this option will take advantage of the gene and mRNA annotations on the sequences as described above. This would typically be annotated genome sequences. See section 19.14.1 for more information. Note that only gene annotations are needed (not mRNA) if you select **Prokaryote** when you click **Next**.

- **Use reference without annotations**. This option is well-suited when using e.g. EST libraries that are not annotated. For this, you need to have a list of sequences. Each sequence in the list will be treated as a "gene" (or "transcript"). Note that the Workbench will treat this as prokaryote (meaning it will not look for new exons (see section 19.14.2) and it will assume that the sequence has no introns). A reference without annotations is typically obtained by importing a fasta file into the Workbench.

Just below these two options, you click to select the reference sequences.



RNA-Seq Analysis

1. Choose where to run
2. Select sequencing reads
3. Set references

Set parameters

Reference

● Use reference with annotations

○ Use reference without annotations

NC_000019_selection

Extend annotated gene regions

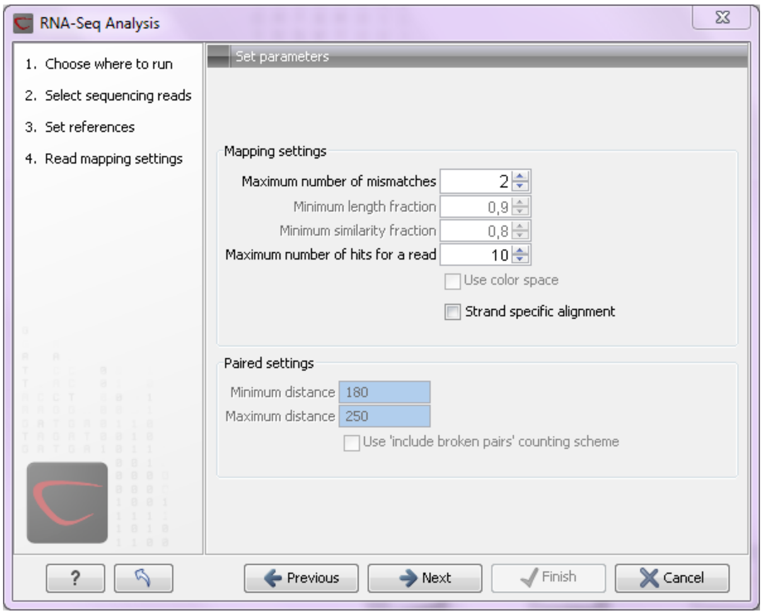
Flanking upstream residues: 0

Flanking downstream residues: 0

Figure 19.108: *Defining a reference genome for RNA-Seq.*

Next, you can choose to extend the region around the gene to include more of the genomic sequence by changing the value in **Flanking upstream/downstream residues**. This also means that you are able to look for new exons before or after the known exons (see section 19.14.2).

When the reference has been defined, click **Next** and you are presented with the dialog shown in figure 19.109.



RNA-Seq Analysis

1. Choose where to run
2. Select sequencing reads
3. Set references
4. Read mapping settings

Set parameters

Mapping settings

Maximum number of mismatches: 2

Minimum length fraction: 0,9

Minimum similarity fraction: 0,8

Maximum number of hits for a read: 10

□ Use color space

□ Strand specific alignment

Paired settings

Minimum distance: 180

Maximum distance: 250

□ Use 'include broken pairs' counting scheme

Figure 19.109: *Defining mapping parameters for RNA-Seq.*

The mapping parameters are:

- **Maximum number of mismatches.** This parameter is available if you use short reads (shorter than 56 nucleotides, except for color space data which are always treated as long reads). This is the maximum number of mismatches to be allowed. Maximum value is 3, except for color space where it is 2.

- **Minimum length fraction**. For long reads, you can specify how much of the sequence should be able to map in order to include it. The default is 0.9 which means that at least 90 % of the bases need to align to the reference.

- **Minimum similarity fraction**. This also applies to long reads and it is used to specify how exact the matching part of the read should be. When using the default setting at 0.8 and the default setting for the length fraction, it means that 90 % of the read should align with 80 % similarity in order to include the read.

- **Maximum number of hits for a read**. A read that matches to more *distinct* places in the references than the 'Maximum number of hits for a read' specified will not be mapped (the notion of *distinct* places is elaborated below). If a read matches to multiple distinct places, but below the specified maximum number, it will be randomly assigned to one of these places. The random distribution is done proportionally to the number of unique matches that the genes to which it matches have, normalized by the exon length (to ensure that genes with no unique matches have a chance of having multi-matches assigned to them, 1 will be used instead of 0, for their count of unique matches). This means that if there are 10 reads that match two different genes with equal exon length, the two reads will be distributed according to the number of unique matches for these two genes. The gene that has the highest number of unique matches will thus get a greater proportion of the 10 reads.

  Places are *distinct* in the references if they are not identical once they have been transferred back to the gene sequences. To exemplify, consider a gene with 10 transcripts and 11 exons, where all transcripts have exon 1, and each of the 10 transcripts have only one of the exons 2 to 11. Exon 1 will be represented 11 times in the references (once for the gene region and once for each of the 10 transcripts). Reads that match to exon 1 will thus match to 11 of the extracted references. However, when transferring the mappings back to the gene it becomes evident that the 11 match places are not distinct but in fact identical. In this case the read will *not* be discarded for exceeding the maximum number of hits limit, but will be mapped. In the RNA-seq action this is algorithmically done by allowing the assembler to return matches that hit in the 'maximum number of hits for a read' *plus* 'the maximum number of transcripts' that the genes have in the specified references. The algorithm post-processes the returned matches to identify the number of distinct matches and only discards a read if this number is above the specified limit. Similarly, when a multi-match read is randomly assigned to one of it's match places, each distinct place is considered only once.

- **Strand-specific alignment**. When this option is checked, the reads will only be mapped in their forward orientation (genes on the minus strand are reverse complemented before mapping). This is useful in places where genes overlap but are on different strands because it is possible to assign the reads to the right gene. Without the strand-specific protocol, this would not be possible (see [Parkhomchuk et al., 2009]).

There is also a checkbox to **Use color space** which is enabled if you have imported a data set from a SOLiD platform containing color space information. Note that color space data is always treated as long reads, regardless of the read length.

Paired data in RNA-Seq

The *CLC Genomics Workbench* supports the use of paired data for RNA-Seq. A combination of single reads and paired reads can also be used. There are three major advantages of using paired data:

- Since the mapped reads span a larger portion of the reference, there will be less non-specifically mapped reads. This means that there is in general a greater accuracy in the expression values.
- This in turn means that there is a greater chance of accurately measuring the expression of transcript splice variants. Since single reads (especially from the short reads platforms) will usually only span one or two exons, there are many cases where the expression splice variants sharing the same exons cannot be determined accurately. With paired reads, more combinations of exons will be identified as unique for a particular splice variant.[2]
- It is possible to detect **Gene fusions** where one read in a pair maps in one gene and the other part maps in another gene. If several reads exhibit the same pattern, there is evidence of a fusion gene.

At the bottom you can specify how **Paired reads** should be handled. You can read more about how paired data is imported and handled in section 19.1. If the sequence list used as input for the assembly contains paired reads, this option will automatically be shown - if it contains single reads, this option will not be shown. Learn more about mapping paired data in section 19.5.3.

When counting the mapped reads to generate expression values, the *CLC Genomics Workbench* needs to decide how to handle paired reads. The standard behavior is this: if two reads map as a pair, the pair is counted as one. If the pair is broken, none of the reads are counted. The reasoning is that something is not right in this case, it could be that the transcripts are not represented correctly on the reference, or there are errors in the data. In general, more confidence is placed with an intact pair. If a combination of paired and single reads are used, "true" single reads will also count as one (the single reads that come from broken pairs will not count).

In some situations it may be too strict to disregard broken pairs. This could be in cases where there is a high degree of variation compared to the reference or where the reference lacks comprehensive transcript annotations. By checking the **Use 'include broken pairs' counting scheme**, both intact and broken pairs are now counted as two. For the broken pairs, this means that each read is counted as one. Reads that are single reads as input are still counted as one.

When looking at the mappings, reads from broken pairs have a darker color than reads that are intact pairs or originally single reads.

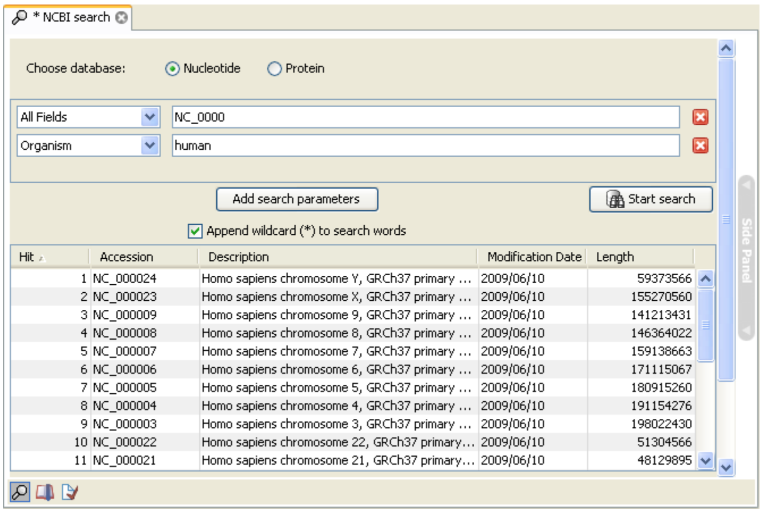
Finding the right reference sequence for RNA-Seq

For prokaryotes, the reference sequence needed for RNA-Seq is quite simple. Either you input a genome annotated with `gene` annotations, or you input a list of genes and select the **Use reference without annotations**.

[2] Note that the *CLC Genomics Workbench* only calculates the expression of the transcripts already annotated on the reference.

For eukaryotes, it is more complex because the Workbench needs to know the intron-exon structure as explained in in the beginning of this section. This means that you need to have a reference genome with annotations of type `mRNA` and `gene` (you can see the annotations of a sequence by opening the annotation table, see section 10.3.1). You can obtain an annotated reference sequence in different ways:

- Download the sequences from NCBI (see section 11.1). Figure 19.110 shows an example of a search for the human refseq chromosomes.
- Retrieve the sequences in annotated format, e.g. GenBank format, and **Import** (📂) them into the Workbench.
- Annotate sequences using a GFF/GTF file containing gene and mRNA annotations (learn more at `http://www.clcbio.com/annotate-with-gff`). If you have the genome in e.g. fasta format, you import this into the Workbench first, and then afterwards you add the annotations from the GFF file. Note that you need to make sure the annotations are the right type. If you use GTF files from Ensembl, these will be fully compatible with the RNA-Seq of *CLC Genomics Workbench*: `ftp://ftp.ensembl.org/pub/current_gtf/` (read more in the Annotate with GFF user manual). Note that GTF files from UCSC cannot be used for RNA-Seq since they do not have information to relate different transcript variants of the same gene.



Choose database: ● Nucleotide ○ Protein

All Fields: NC_0000

Organism: human

☑ Append wildcard (*) to search words

Hit	Accession	Description	Modification Date	Length
1	NC_000024	Homo sapiens chromosome Y, GRCh37 primary ...	2009/06/10	59373566
2	NC_000023	Homo sapiens chromosome X, GRCh37 primary ...	2009/06/10	155270560
3	NC_000009	Homo sapiens chromosome 9, GRCh37 primary ...	2009/06/10	141213431
4	NC_000008	Homo sapiens chromosome 8, GRCh37 primary ...	2009/06/10	146364022
5	NC_000007	Homo sapiens chromosome 7, GRCh37 primary ...	2009/06/10	159138663
6	NC_000006	Homo sapiens chromosome 6, GRCh37 primary ...	2009/06/10	171115067
7	NC_000005	Homo sapiens chromosome 5, GRCh37 primary ...	2009/06/10	180915260
8	NC_000004	Homo sapiens chromosome 4, GRCh37 primary ...	2009/06/10	191154276
9	NC_000003	Homo sapiens chromosome 3, GRCh37 primary ...	2009/06/10	198022430
10	NC_000022	Homo sapiens chromosome 22, GRCh37 primary...	2009/06/10	51304566
11	NC_000021	Homo sapiens chromosome 21, GRCh37 primary...	2009/06/10	48129895

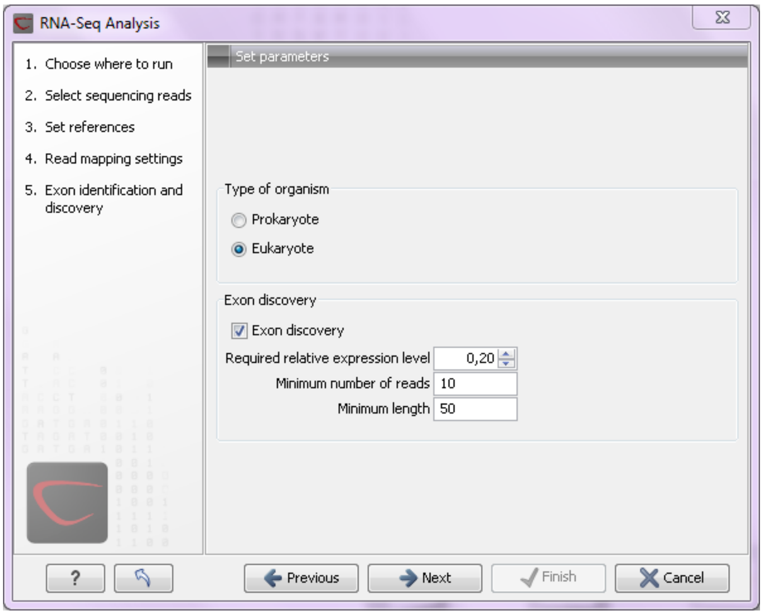
Figure 19.110: *Downloading the human genome from refseq.*

19.14.2 Exon identification and discovery

Clicking **Next** will show the dialog in figure 19.111.

The choice between **Prokaryote** and **Eukaryote** is basically a matter of telling the Workbench whether you have introns in your reference. In order to select **Eukaryote**, you need to have reference sequences with annotations of the type mRNA (this is the way the Workbench expects exons to be defined - see section 19.14).

Here you can specify the settings for discovering novel exons. The mapping will be performed against the entire gene, and by analyzing the reads located between known exons, the *CLC*



Type of organism

○ Prokaryote

● Eukaryote

Exon discovery

☑ Exon discovery

Required relative expression level 0,20

Minimum number of reads 10

Minimum length 50

Figure 19.111: *Exon identification and discovery.*

Genomics Workbench is able to report new exons. A new exon has to fulfill the parameters you set:

- **Required relative expression level**. This is the expression level relative to the rest of the gene. A value of 20% means that the expression level of the new exon has to be at least 20% of that of the known exons of this gene.
- **Minimum number of reads**. While the previous option asks for the percentage relative to the general expression level of the gene, this option requires an absolute value. Just a few matching reads will already be considered to be a new exon for genes with low expression levels. This is avoided by setting a minimum number of reads here.
- **Minimum length**. This is the minimum length of an exon. There has to be overlapping reads for the whole minimum length.

Figure 19.112 shows an example of a putative exon.

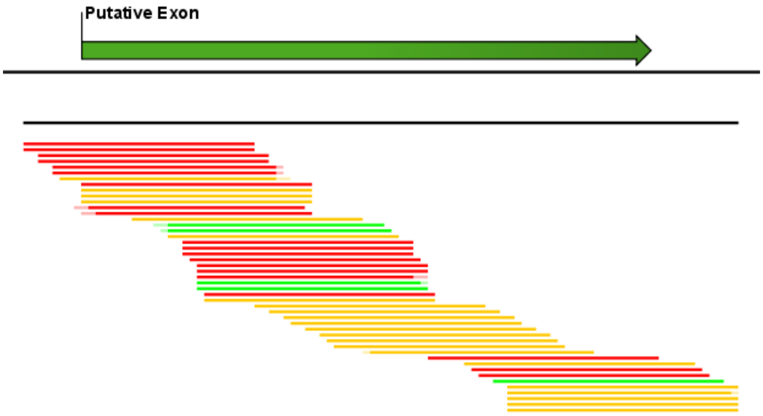
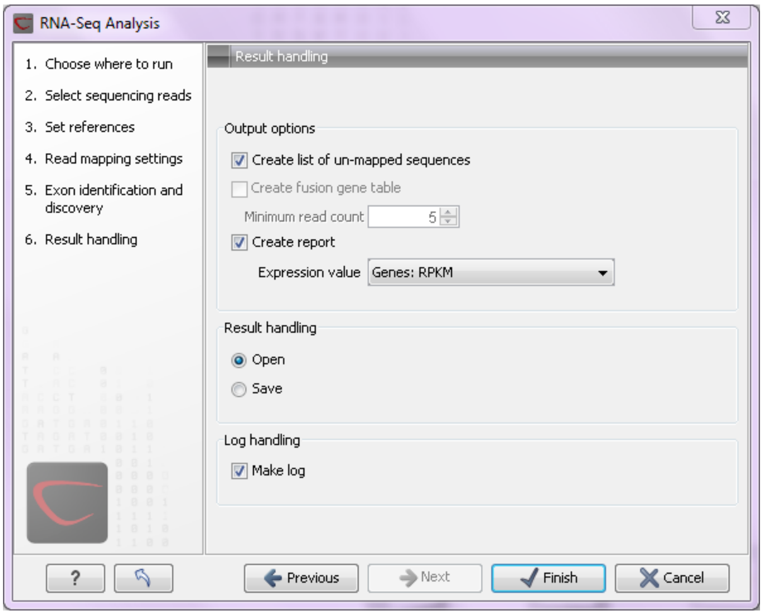


Figure 19.112: *A putative exon has been identified.*

19.14.3 RNA-Seq output options

Clicking **Next** will allow you to specify the output options as shown in figure 19.113.



Output options

☑ Create list of un-mapped sequences

□ Create fusion gene table

Minimum read count: 5

☑ Create report

Expression value: Genes: RPKM

Result handling

● Open

○ Save

Log handling

☑ Make log

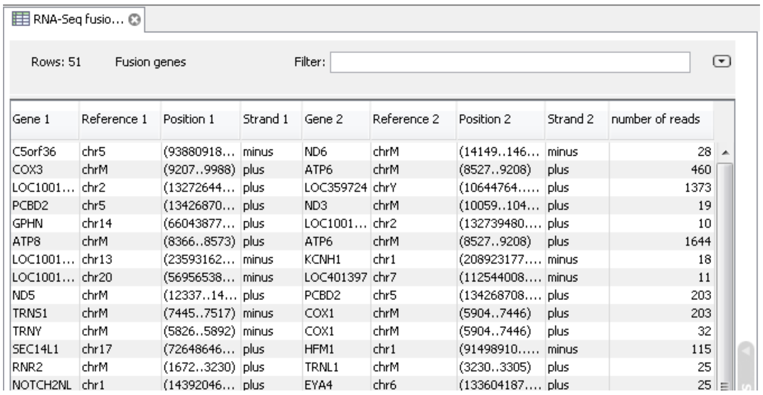
Figure 19.113: *Selecting the output of the RNA-Seq analysis.*

The standard output is a table showing statistics on each gene and the option to open the mapping (see more below). Furthermore, the expression of individual transcripts is reported (for eukaryotes). The expression measure used for further analysis can be specified as well. Per default it is set to Genes RPKM. This can also be changed at a later point (see below).

Furthermore, you can choose to create a sequence list of the **non-mapped sequences**. This could be used to do *de novo* assembly and perform BLAST searches to see if you can identify new genes or at least further investigate the results.

Gene fusion reporting

When using paired data, there is also an option to create a table summarizing the evidence for gene fusions. And example is shown in figure 19.114.



Rows: 51 Fusion genes Filter:

Gene 1	Reference 1	Position 1	Strand 1	Gene 2	Reference 2	Position 2	Strand 2	number of reads
C5orf36	chr5	(93880918...	minus	ND6	chrM	(14149..146...	minus	28
COX3	chrM	(9207..9988)	plus	ATP6	chrM	(8527..9208)	plus	460
LOC1001...	chr2	(13272644...	plus	LOC359724	chrY	(10644764.....	plus	1373
PCBD2	chr5	(13426870...	plus	ND3	chrM	(10059..104...	plus	19
GPHN	chr14	(66043877...	plus	LOC1001...	chr2	(132739480....	plus	10
ATP8	chrM	(8366..8573)	plus	ATP6	chrM	(8527..9208)	plus	1644
LOC1001...	chr13	(23593162...	minus	KCNH1	chr1	(208923177....	minus	18
LOC1001...	chr20	(56956538...	minus	LOC401397	chr7	(112544008....	minus	11
ND5	chrM	(12337..14...	plus	PCBD2	chr5	(134268708....	plus	203
TRNS1	chrM	(7445..7517)	minus	COX1	chrM	(5904..7446)	plus	203
TRNY	chrM	(5826..5892)	minus	COX1	chrM	(5904..7446)	plus	32
SEC14L1	chr17	(72648646...	plus	HFM1	chr1	(91498910.....	minus	115
RNR2	chrM	(1672..3230)	plus	TRNL1	chrM	(3230..3305)	plus	25
NOTCH2NL	chr1	(14392046...	plus	EYA4	chr6	(133604187....	plus	25

Figure 19.114: *An example of a gene fusion table.*

The table includes the following columns for each part of the pair:

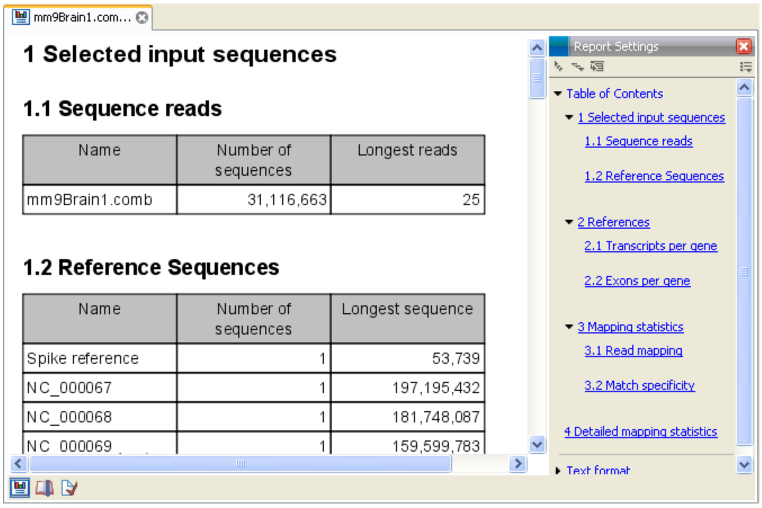
- **Gene**. The name of the gene.
- **Reference**. The name of the reference sequence (typically the chromosome name).
- **Position**. The position of the gene.
- **Strand**. The strand of the gene.

Most importantly, the table lists the number of read pairs that are supporting the combination of genes listed. The threshold for when a combination of genes should be reported in the table can be set in the RNA-Seq dialog in figure 19.113. The default value is 5.

Note that the reporting of gene fusions is very simple and should be analyzed in much greater detail before any evidence of gene fusions can be verified. The table should be considered more of a pointer to genes to explore rather than evidence of gene fusions.

RNA-Seq report

In addition, there is an option to **Create report**. This will create a report as shown in figure 19.115.



1 Selected input sequences

1.1 Sequence reads

Name	Number of sequences	Longest reads
mm9Brain1.comb	31,116,663	25

1.2 Reference Sequences

Name	Number of sequences	Longest sequence
Spike reference	1	53,739
NC_000067	1	197,195,432
NC_000068	1	181,748,087
NC_000069	1	159,599,783

Figure 19.115: *Report of an RNA-Seq run.*

The report contains the following information:

- **Sequence reads**. Information about the number of reads.
- **Reference sequences**. Information about the reference sequences used and their lengths and the total number of genes found in the reference.
- **Transcripts per gene**. A graph showing the number of transcripts per gene. For eukaryotes, this will be equivalent to the number of mRNA annotations per gene annotation.
- **Exons per gene**. A graph showing the number of exons per gene.
- **Exons per transcript**. A graph showing the number of exons per transcript.
- **Read mapping**. Shows statistics on:

  - **Mapped reads**. This number is divided into uniquely and non-specifically mapped reads (see the point below on match specificity for details).
  - **Unmapped reads**.
  - **Total reads**. This is the number of reads used as input.

- **Paired reads**. (Only included if paired reads are used). Shows the number of reads mapped in pairs, the number of reads in broken pairs and the number of unmapped reads.

- **Match specificity**. Shows a graph of the number of match positions for the reads. Most reads will be mapped 0 or 1 time, but there will also be reads matching more than once in the reference. This depends on the **Maximum number of hits for a read** setting in figure19.108. Note that the number of reads that are mapped 0 times includes both the number of reads that cannot be mapped at all and the number of reads that matches to more than the 'Maximum number of hits for a read' parameter that you set in the second wizard step. If paired reads are used, a separate graph is produced for that part of the data.

- **Paired distance**. (Only included if paired reads are used). Shows a graph of the distance between mapped reads in pairs.

- **Detailed mapping statistics**. This table divides the reads into the following categories.

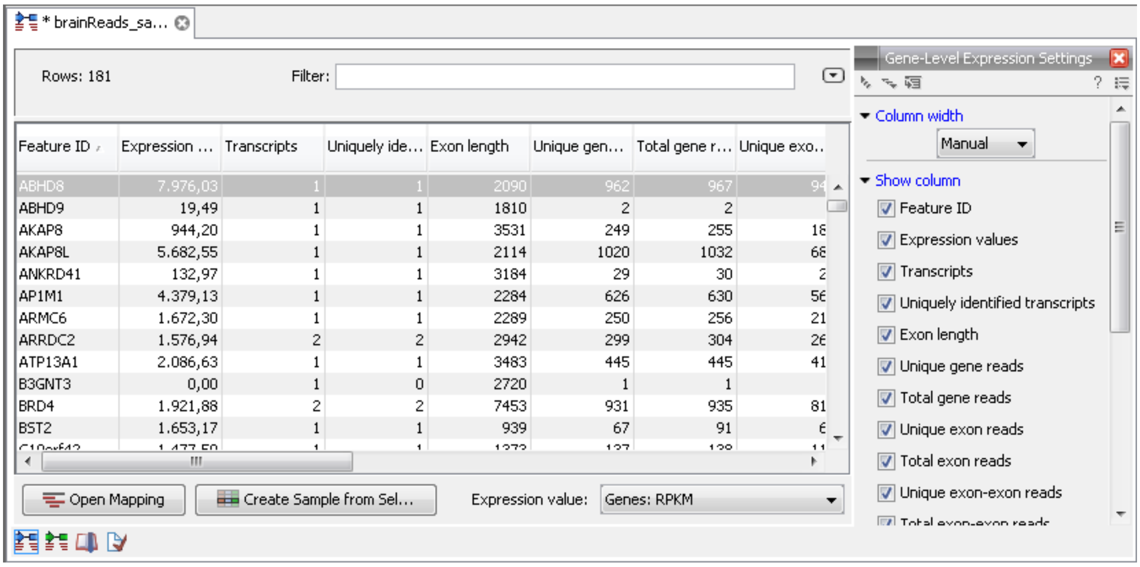
  - **Exon-exon reads**. Reads that overlap two exons as specified in figure 19.111.
  - **Exon-intron reads**. Reads that span both an exon and an intron. If you have many of these reads, it could indicate a low splicing-efficiency or that a number of splice variants are not annotated on your reference.
  - **Total exon reads**. Number of reads that fall entirely within an exon or in an exon-exon junction.
  - **Total intron reads**. Reads that fall entirely within an intron or in the gene's flanking regions.
  - **Total gene reads**. All reads that map to the gene and it's flanking regions. This is the *mapped reads* number used for calculating RPKM, see definition below.

  For each category, the number of uniquely and non-specifically mapped reads are listed as well as the relative fractions. Note that all this detailed information is also available on the individual gene level in the RNA-Seq table ()(see below). When the input data is a combination of paired and single reads, the mapping statistics will be divided into two parts.

Note that the report can be exported in pdf or Excel format.

19.14.4 Interpreting the RNA-Seq analysis result

The main result of the RNA-Seq is the reporting of expression values which is done on both the gene and the transcript level (only eukaryotes).



Rows: 181 Filter:

Feature ID	Expression ...	Transcripts	Uniquely ide...	Exon length	Unique gen...	Total gene r...	Unique exo..
ABHD8	7.976,03	1	1	2090	962	967	94
ABHD9	19,49	1	1	1810	2	2	
AKAP8	944,20	1	1	3531	249	255	18
AKAP8L	5.682,55	1	1	2114	1020	1032	68
ANKRD41	132,97	1	1	3184	29	30	2
AP1M1	4.379,13	1	1	2284	626	630	56
ARMC6	1.672,30	1	1	2289	250	256	21
ARRDC2	1.576,94	2	2	2942	299	304	26
ATP13A1	2.086,63	1	1	3483	445	445	41
B3GNT3	0,00	1	0	2720	1	1	
BRD4	1.921,88	2	2	7453	931	935	81
BST2	1.653,17	1	1	939	67	91	6

Expression value: Genes: RPKM

Column width
Manual

Show column
☑ Feature ID
☑ Expression values
☑ Transcripts
☑ Uniquely identified transcripts
☑ Exon length
☑ Unique gene reads
☑ Total gene reads
☑ Unique exon reads
☑ Total exon reads
☑ Unique exon-exon reads
☑ Total exon-exon reads

Figure 19.116: *A subset of a result of an RNA-Seq analysis on the gene level. Not all columns are shown in this figure*

Gene-level expression

When you open the result of an RNA-Seq analysis, it starts in the gene-level view as shown in figure 19.116.

The table summarizes the read mappings that were obtained for each gene (or reference). The following information is available in this table:

- **Feature ID**. This is the name of the gene.
- **Expression values**. This is based on the expression measure chosen in figure 19.113.
- **Transcripts**. The number of transcripts based on the mRNA annotations on the reference. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).
- **Detected transcripts**. The number of transcripts which have reads assigned (see the description of transcript-level expression below).
- **Exon length**. The total length of all exons (not all transcripts).
- **Unique gene reads**. This is the number of reads that match uniquely to the gene.
- **Total gene reads**. This is all the reads that are mapped to this gene — both reads that map uniquely to the gene and reads that matched to more positions in the reference (but fewer than the 'Maximum number of hits for a read' parameter) which were assigned to this gene.
- **Unique exon reads**. The number of reads that match uniquely to the exons (including the exon-exon and exon-intron junctions).
- **Total exon reads**. Number of reads mapped to this gene that fall entirely within an exon or in exon-exon or exon-intron junctions. As for the 'Total gene reads' this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon of this gene.

- **Unique exon-exon reads**. Reads that uniquely match across an exon-exon junction of the gene (as specified in figure 19.111). The read is only counted once even though it covers several exons.

- **Total exon-exon reads**. Reads that match across an exon-exon junction of the gene (as specified in figure 19.111). As for the 'Total gene reads' this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon-exon junction of this gene.

- **Unique intron-exon reads**. Reads that uniquely map across an exon-intron boundary. If you have many of these reads, it could indicate that a number of splice variants are not annotated on your reference.

- **Total intron-exon reads**. Reads that map across an exon-intron boundary. As for the 'Total gene reads' this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon-intron junction of this gene. If you have many of these reads, it could indicate that a number of splice variants are not annotated on your reference.

- **Exons**. The number of exons based on the mRNA annotations on the reference. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).

- **Putative exons**. The number of new exons discovered during the analysis (see more in section 19.14.2).

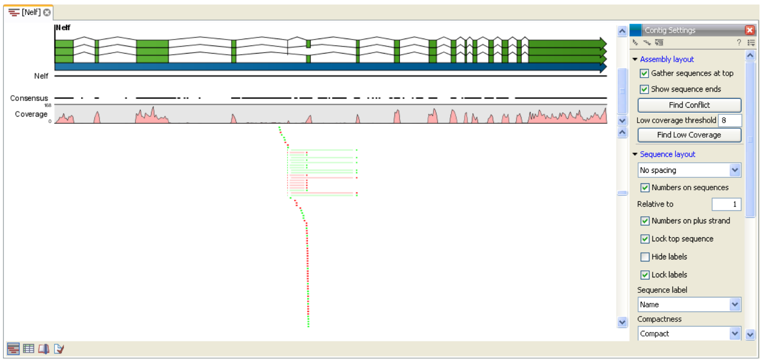
- **RPKM**. This is the expression value measured in RPKM [Mortazavi et al., 2008]: $RPKM = \frac{\textit{total exon reads}}{\textit{mapped reads(millions)} \times \textit{exon length (KB)}}$. See exact definition below. Even if you have chosen the RPKM values to be used in the **Expression values** column, they will also be stored in a separate column. This is useful to store the RPKM if you switch the expression measure. See more in section 19.14.4.

- **Median coverage**. This is the median coverage for all exons (for all reads - not only the unique ones). Reads spanning exon-exon boundaries are not included.

- **Chromosome region start**. Start position of the annotated gene.

- **Chromosome region end**. End position of the annotated gene.

Double-clicking any of the genes will open the mapping of the reads to the reference (see figure 19.117).



Assembly layout
☑ Gather sequences at top
☑ Show sequence ends
Find Conflict
Low coverage threshold 8
Find Low Coverage

Sequence layout
No spacing
☑ Numbers on sequences
Relative to 1
☑ Numbers on plus strand
☑ Lock top sequence
□ Hide labels
☑ Lock labels
Sequence label: Name
Compactness: Compact

Figure 19.117: *Opening the mapping of the reads. Zoomed out to provide a better overview.*

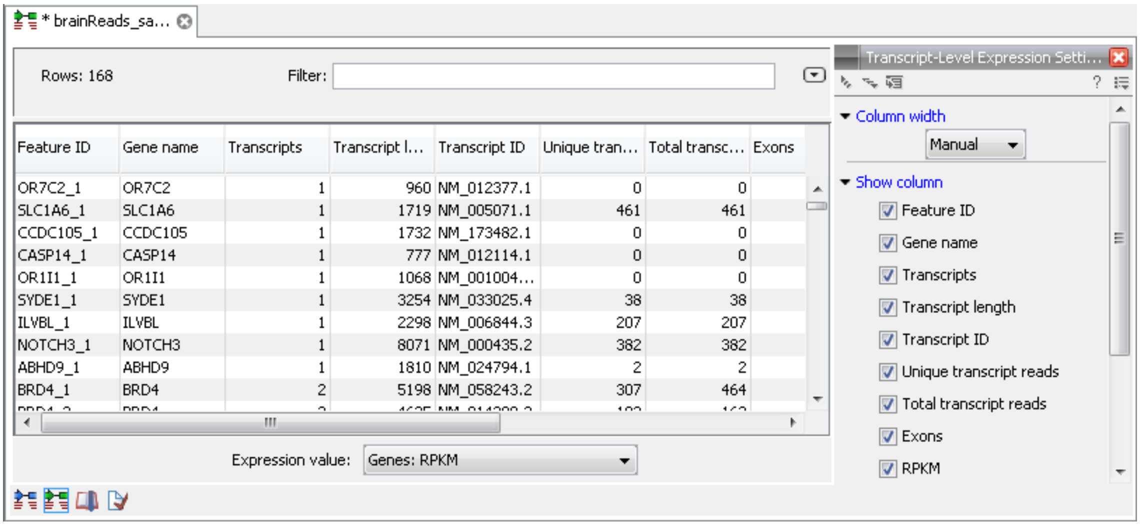
Reads spanning two exons are shown with a faded line between each end as shown in figure 19.117.

At the bottom of the table you can change the expression measure. Simply select another value in the drop-down list. The expression measure chosen here is the one used for further analysis. When setting up an experiment, you can specify an expression value to apply to all samples in the experiment.

The RNA-Seq analysis result now represents the expression values for the sample, and it can be further analyzed using the various tools described in chapter 20.

Transcript-level expression

In order to switch to the transcript-level expression, click the **Transcript-level expression** (![]) button at the bottom of the view. You will now see a view as shown in figure 19.118.



Rows: 168

Filter:

Feature ID	Gene name	Transcripts	Transcript l...	Transcript ID	Unique tran...	Total transc...	Exons
OR7C2_1	OR7C2	1	960	NM_012377.1	0	0	
SLC1A6_1	SLC1A6	1	1719	NM_005071.1	461	461	
CCDC105_1	CCDC105	1	1732	NM_173482.1	0	0	
CASP14_1	CASP14	1	777	NM_012114.1	0	0	
OR1I1_1	OR1I1	1	1068	NM_001004...	0	0	
SYDE1_1	SYDE1	1	3254	NM_033025.4	38	38	
ILVBL_1	ILVBL	1	2298	NM_006844.3	207	207	
NOTCH3_1	NOTCH3	1	8071	NM_000435.2	382	382	
ABHD9_1	ABHD9	1	1810	NM_024794.1	2	2	
BRD4_1	BRD4	2	5198	NM_058243.2	307	464	

Expression value: Genes: RPKM

Transcript-Level Expression Setti...

Column width

Manual

Show column

☑ Feature ID
☑ Gene name
☑ Transcripts
☑ Transcript length
☑ Transcript ID
☑ Unique transcript reads
☑ Total transcript reads
☑ Exons
☑ RPKM

Figure 19.118: *A subset of a result of an RNA-Seq analysis on the transcript level. Not all columns are shown in this figure*

The following information is available in this table:

- **Feature ID**. This is the gene name with a number appended to differentiate between transcripts.
- **Expression values**. This is based on the expression measure chosen in figure 19.113.
- **Transcripts**. The number of transcripts based on the mRNA annotations on the reference. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).
- **Transcript length**. The total length of all exons of that particular transcript.
- **Transcript ID**. This information is retrieved from transcript_ID key on the mRNA annotation.
- **Unique transcript reads**. This is the number of reads in the mapping for the gene that are uniquely assignable to the transcript. This number is calculated after the reads have been mapped and both single and multi-hit reads from the read mapping may be unique transcript reads.

- **Total transcript reads**. Once the 'Unique transcript read's have been identified and their counts calculated for each transcript, the remaining (non-unique) transcript reads are assigned randomly to one of the transcripts to which they match. The 'Total transcript reads' counts are the total number of reads that are assigned to the transcript once this random assignment has been done. As for the random assignment of reads among genes, the random assignment of reads within a gene but among transcripts, is done proportionally to the 'unique transcript counts' normalized by transcript length, that is, using the RPKM (see the description of the 'Maximum number of hits for a read' option', 19.14.1). Unique transcript counts of 0 are not replaced by 1 for this proportional assignment of non-unique reads among transcripts.

- **Exons**. The number of exons for this transcript. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).

- **RPKM**. The RPKM value for the transcript, that is, the number of reads assigned to the transcript divided by the transcript length and normalized by 'Mapped reads' (see below).

- **Relative RPKM**. The RPKM value for the transcript divided by the maximum of the RPKM values for transcripts for this gene.

- **Chromosome region start**. Start position of the annotated gene.

- **Chromosome region end**. End position of the annotated gene.

Definition of RPKM

RPKM, Reads Per Kilobase of exon model per Million mapped reads, is defined in this way [Mortazavi et al., 2008]: $RPKM = \frac{\textit{total exon reads}}{\textit{mapped reads(millions)}\times\textit{exon length (KB)}}$.

Total exon reads This is the number in the column with header **Total exon reads** in the row for the gene. This is the number of reads that have been mapped to a region in which an exon is annotated for the gene or across the boundaries of two exons or an intron and an exon for an annotated transcript of the gene. For eukaryotes, exons and their internal relationships are defined by annotations of type mRNA.

Exon length This is the number in the column with the header **Exon length** in the row for the gene, divided by 1000. This is calculated as the sum of the lengths of all exons annotated for the gene. Each exon is included only once in this sum, even if it is present in more annotated transcripts for the gene. Partly overlapping exons will count with their full length, even though they share the same region.

Mapped reads The sum of all the numbers in the column with header **Total gene reads**. The **Total gene reads** for a gene is the total number of reads that after mapping have been mapped to the region of the gene. Thus this includes all the reads uniquely mapped to the region of the gene as well as those of the reads which match in more places (below the limit set in the dialog in figure 19.108) that have been allocated to this gene's region. A gene's region is that comprised of the flanking regions (if it was specified in figure 19.108), the exons, the introns and across exon-exon boundaries of all transcripts annotated for the gene. Thus, the sum of the total gene reads numbers is the number of mapped reads for the sample (you can find the number in the RNA-Seq report).

19.15 Expression profiling by tags

Expression profiling by tags, also known as *tag profiling* or *tag-based transcriptomics*, is an extension of Serial analysis of gene expression (SAGE) using next-generation sequencing technologies. With respect to sequencing technology it is similar to RNA-seq (see section 19.14), but with tag profiling, you do not sequence the mRNA in full length. Instead, small tags are extracted from each transcript, and these tags are then sequenced and counted as a measure of the abundance of each transcript. In order to tell which gene's expression a given tag is measuring, the tags are often compared to a virtual tag library. This consists of the 'virtual' tags that would have been extracted from an annotated genome or a set of ESTs, had the same protocol been applied to these. For a good introduction to tag profiling including comparisons with different micro array platforms, we refer to ['t Hoen et al., 2008]. For more in-depth information, we refer to [Nielsen, 2007].

Figure 19.119 shows an example of the basic principle behind tag profiling. There are variations of this concept and additional details, but this figure captures the essence of tag profiling, namely the extraction of a tag from the mRNA based on restriction cut sites.

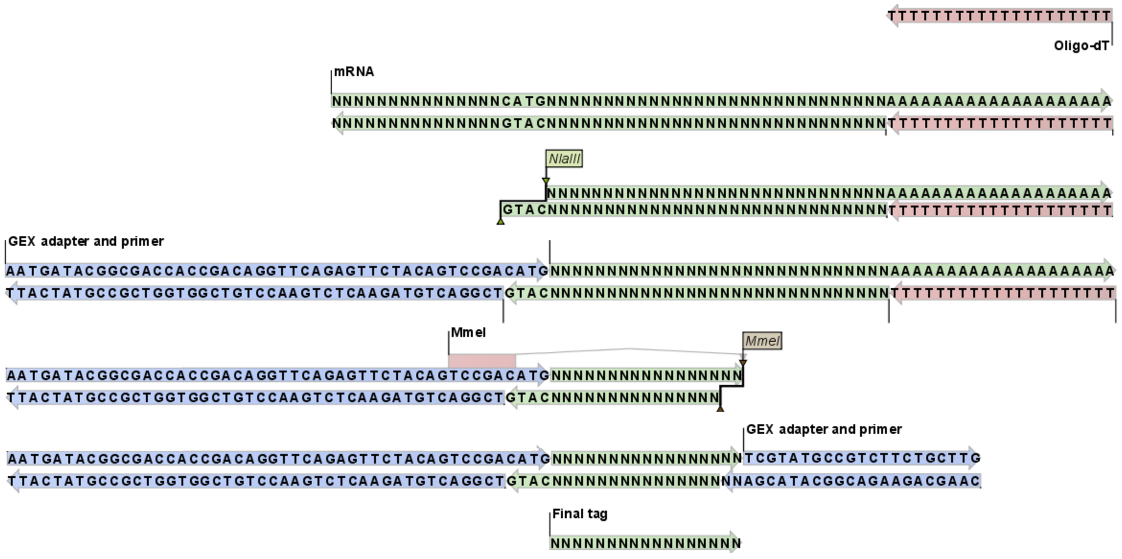


Figure 19.119: *An example of the tag extraction process. 1+2. Oligo-dT attached to a magnetic bead is used to trap mRNA. 3. The enzyme* NlaIII *cuts at CATG sites and the fragments not attached to the magnetic bead are removed. 4. An adapter is ligated to the GTAC overang. 5. The adapter includes a recognition site for* MmeI *which cuts 17 bases downstream. 6. Another adapter is added and the sequence is now ready for amplification and sequencing. 7. The final tag is 17 bp. The example is inspired by ['t Hoen et al., 2008].*

The *CLC Genomics Workbench* supports the entire tag profiling data analysis work flow following the sequencing:

- Extraction of tags from the raw sequencing reads (tags from different samples are often barcoded and sequenced in one pool).

- Counting tags including a sequencing-error correction algorithm.
- Creating a virtual tag list based on an annotated reference genome or an EST-library.
- Annotating the tag counts with gene names from the virtual tag list.

Each of the steps in the work flow are described in details below.

19.15.1 Extract and count tags

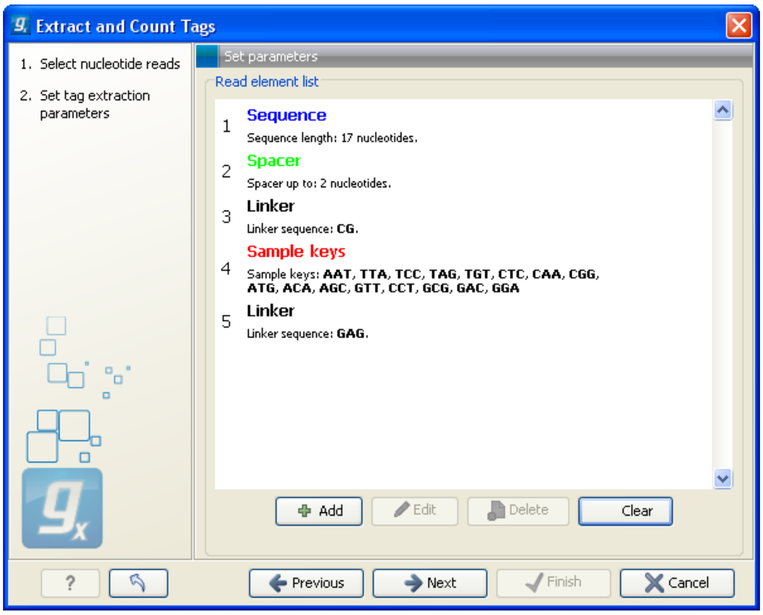
First step in the analysis is to import the data (see section 19.1).

The next step is to extract the tags and count them:

> **Toolbox** | **High-throughput Sequencing** ( ) | **Expression Profiling by Tags** ( ) | **Extract and Count Tags** ( )

This will open a dialog where you select the reads that you have imported. Click **Next** when the sequencing data is listed in the right-hand side of the dialog.

This dialog is where you define the elements in your reads. An example is shown in figure 19.120.



1 **Sequence**
Sequence length: 17 nucleotides.

2 **Spacer**
Spacer up to: 2 nucleotides.

3 **Linker**
Linker sequence: **CG**.

4 **Sample keys**
Sample keys: **AAT, TTA, TCC, TAG, TGT, CTC, CAA, CGG, ATG, ACA, AGC, GTT, CCT, GCG, GAC, GGA**

5 **Linker**
Linker sequence: **GAG**.

Figure 19.120: *Defining the elements that make up your reads.*

By defining the order and size of each element, the Workbench is now able to both separate samples based on bar codes and extract the tag sequence (i.e. removing linkers, bar codes etc). The elements available are:

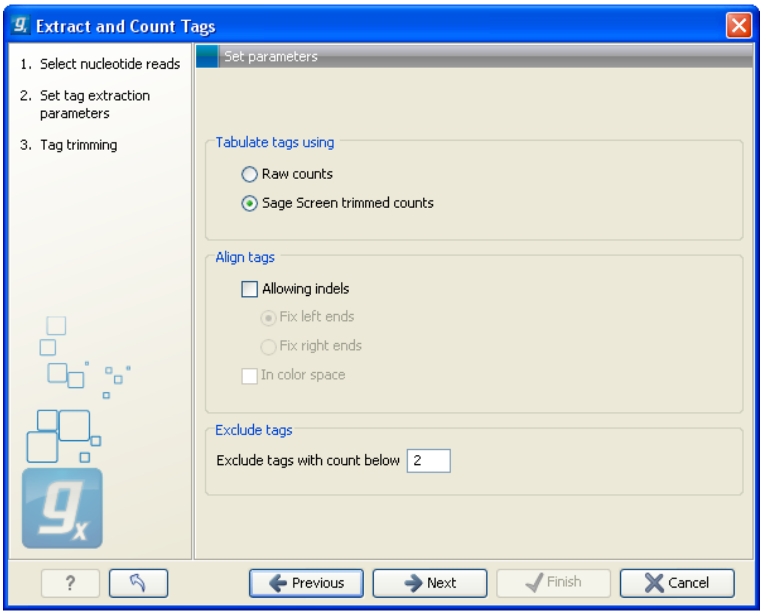
Sequence This is the part of the read that you want to use as your final tag for counting and annotating. If you have tags of varying lengths, add a spacer afterwards (see below).

Sample keys Here you input a comma-separated list of the sample keys used for identifying the samples (also referred to as "bar codes"). If you have not pooled and bar coded your data, simply omit this element.

Linker This is a known sequence that you know should be present and do not want to be included in your final tag.

Spacer This is also a sequence that you do not want to include in your final tag, but whereas the linker is defined by its sequence, the spacer is defined by its length. Note that the length defines the maximum length of the spacer. Often not all tags will be exactly the same length, and you can use this spacer as a buffer for those tags that are longer than what you have defined as your sequence. In the example in figure 19.120, the tag length is 17 bp, but a spacer is added to allow tags up to 19 bp. Note that the part of the read that is extracted and used as the final tag does not include the spacer sequence. In this way you homogenize the tag lengths which is usually desirable because you want to count short and long tags together.

When you have set up the right order of your elements, click **Next** to set parameters for counting tags as shown in figure 19.121.



Tabulate tags using
○ Raw counts
● Sage Screen trimmed counts

Align tags
□ Allowing indels
● Fix left ends
○ Fix right ends
□ In color space

Exclude tags
Exclude tags with count below 2

Figure 19.121: *Setting parameters for counting tags.*

At the top, you can specify how to tabulate (i.e. count) the tags:

Raw counts This will produce the count for each tag in the data.

Sage Screen trimmed counts This will produce trimmed tag counts. The trimmed tag counts are obtained by applying an implementation of the SAGEscreen method ([Akmaev and Wang, 2004]) to the raw tag counts. In this procedure, raw counts are trimmed using probabilistic reasoning. In this procedure, if a tag with low count has a neighboring tag with high count, and it is likely, based on the estimated mutation rate, that the low count tags have arisen through sequencing errors of the tags with higher count, the count of the less abundant tag will be attributed to the higher abundant neighboring tag. The implementation of the SAGEscreen method is highly efficient and provides considerable speed and memory improvements.

Next, you can specify additional parameters for the alignment that takes place when the tags are tabulated:

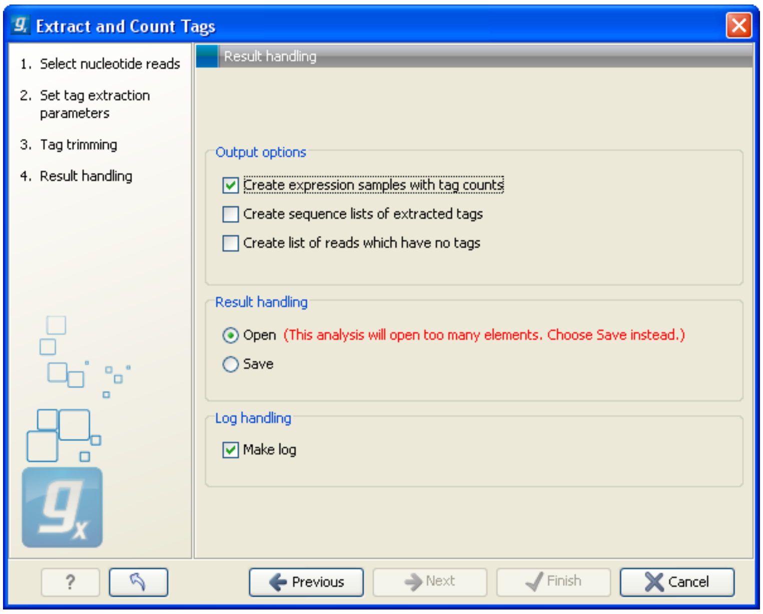
Allowing indels Ticking this box means that, when SAGEscreen is applied, neighboring tags will, in addition to tags which differ by nucleotide substitutions, also include tags with insertion or deletion differences.

Color space This option is only available if you use data generated on the SOLiD platform. Checking this option will perform the alignment in color space which is desirable because sequencing errors can be corrected. Learn more about color space in section 19.8.

At the bottom you can set a minimum threshold for tags to be reported. Although the SAGEscreen trimming procedure will reduce the number of erroneous tags reported, the procedure only handles tags that are neighbors of more abundant tags. Because of sequencing errors, there will be some tags that show extensive variation. There will by chance only be a few copies of these tags, and you can use the minimum threshold option to simply discard tags. The default value is two which means that tags only occurring once are discarded. This setting is a trade-off between removing bad-quality tags and still keeping tags with very low expression (the ability to measure low levels of mRNA is one of the advantages of tag profiling over for example micro arrays ['t Hoen et al., 2008]).

Note! If more samples are created, SAGEscreen and the minimum threshold cut-offs will be applied to the cumulated counts (i.e. all tags for all samples).

Clicking **Next** allows you to specify the output of the analysis as shown in figure 19.122.



Extract and Count Tags

1. Select nucleotide reads
2. Set tag extraction parameters
3. Tag trimming
4. Result handling

Result handling

Output options

☑ Create expression samples with tag counts

□ Create sequence lists of extracted tags

□ Create list of reads which have no tags

Result handling

● Open (This analysis will open too many elements. Choose Save instead.)

○ Save

Log handling

☑ Make log

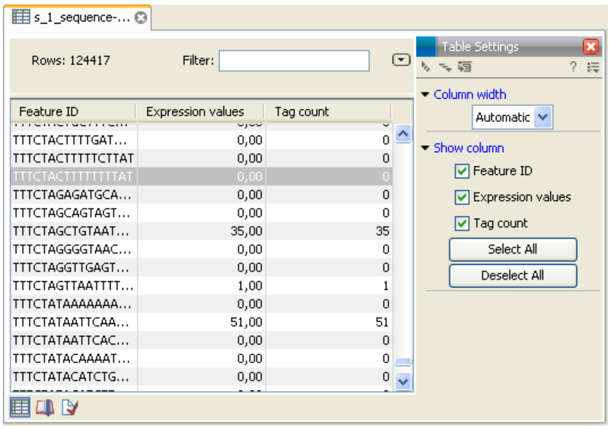
Figure 19.122: *Output options.*

The options are:

Create expression samples with tag counts This is the primary result showing all the tags and respective counts (an example is shown in figure 19.123). For each sample defined via the bar codes, there will be an expression sample like this. Note that all samples have the same list of tags, even if the tag is not present in the given sample (i.e. there will be tags with count 0 as shown in figure 19.123). The expression samples can be used in further analysis by the expression analysis tools (see chapter 20).

Create sequence lists of extracted tags This is a simple sequence list of all the tags that were extracted. The list is simple with no counts or additional information.

Create list of reads which have no tags This list contains the reads from which a tag could not be extracted. This is most likely bad quality reads with sequencing errors that make them impossible to group by their bar codes. It can be useful for troubleshooting if the amount of real tags is smaller than expected.



s_1_sequence-...

Rows: 124417 Filter:

Feature ID	Expression values	Tag count
TTTCTACTTTTGAT...	0,00	0
TTTCTACTTTTTCTTAT	0,00	0
TTTCTACTTTTTTTTAT	0,00	0
TTTCTAGAGATGCA...	0,00	0
TTTCTAGCAGTAGT...	0,00	0
TTTCTAGCTGTAAT...	35,00	35
TTTCTAGGGGTAAC...	0,00	0
TTTCTAGGTTGAGT...	0,00	0
TTTCTAGTTAATTTT...	1,00	1
TTTCTATAAAAAAA...	0,00	0
TTTCTATAATTCAA...	51,00	51
TTTCTATAATTCAC...	0,00	0
TTTCTATACAAAAT...	0,00	0
TTTCTATACATCTG...	0,00	0

Table Settings

Column width: Automatic

Show column

☑ Feature ID

☑ Expression values

☑ Tag count

Figure 19.123: *The tags have been extracted and counted.*

Finally, a log can be shown of the extraction and count process. The log gives useful information such as the number of tags in each sample and the number of reads without tags.

19.15.2 Create virtual tag list

Before annotating the tag sample ( ) created above, you need to create a so-called *virtual tag list*. The list is created based on an annotated genome or a list of ESTs, and it represents the tags that you would expect to find in your experimental data (given the reference genome or EST list reflects your sample). To create the list, you specify the restriction enzyme and tag length to be used for creating the virtual list.

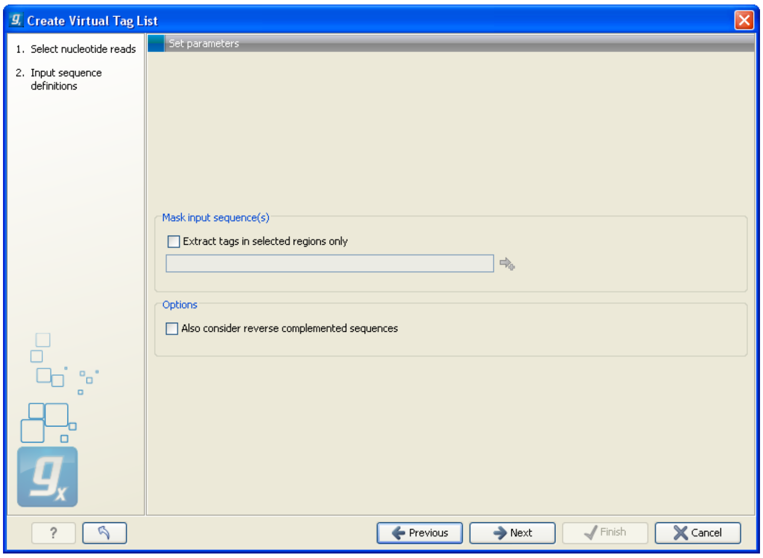
The virtual tag list can be saved and used to annotate experiments made from tag-based expression samples as shown in section 19.15.3.

To create the list:

Toolbox | **High-throughput Sequencing ( )** | **Expression Profiling by Tags ( )** | **Create Virtual Tag List ( )**

This will open a dialog where you select one or more annotated genomic sequences or a list of ESTs. Click **Next** when the sequences are listed in the right-hand side of the dialog.

This dialog is where you specify the basis for extracting the virtual tags (see figure 19.124).



Mask input sequence(s)

□ Extract tags in selected regions only

Options

□ Also consider reverse complemented sequences

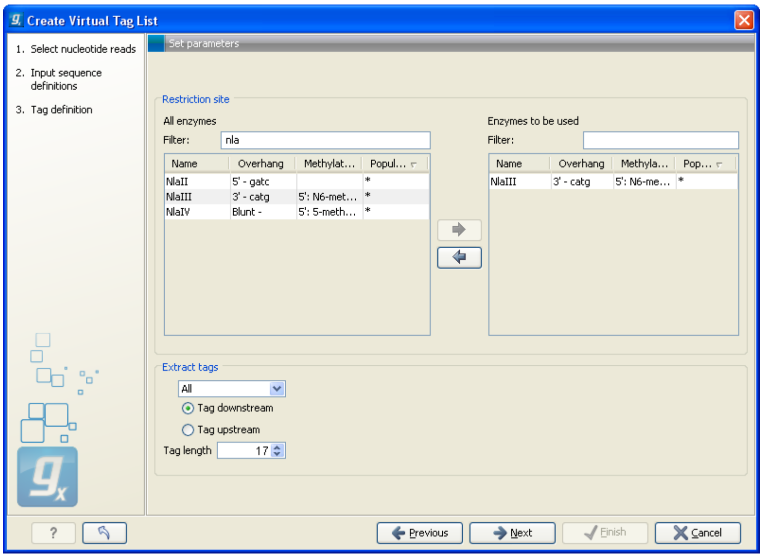
Figure 19.124: *The basis for the extraction of reads.*

At the top you can choose to extract tags based on annotations on your sequences by checking the **Extract tags in selected areas only** option. This option is applicable if you are using annotated genomes (e.g. Refseq genomes). Click the small button (➡) to the right to display a dialog showing all the annotation types in your sequences. Select the annotation type representing your transcripts (usually *mRNA* or *Gene*). The sequence fragments covered by the selected annotations will then be extracted from the genomic sequence and used as basis for creating the virtual tag list.

If you use a sequence list where each sequence represents your transcript (e.g. an EST library), you should not check the **Extract tags in selected areas only** option.

Below, you can choose to include the reverse complement for creating virtual tags. This is mainly used if there is uncertainty about the orientation of sequences in an EST library.

Clicking **Next** allows you to specify enzymes and tag length as shown in figure 19.125.



Restriction site

All enzymes

Filter: nla

Name	Overhang	Methylat...	Popul...
NlaII	5' - gatc		*
NlaIII	3' - catg	5': N6-met...	*
NlaIV	Blunt -	5': 5-meth...	*

Enzymes to be used

Filter:

Name	Overhang	Methyla...	Pop...
NlaIII	3' - catg	5': N6-me...	*

Extract tags

All

● Tag downstream

○ Tag upstream

Tag length 17

Figure 19.125: *Defining restriction enzyme and tag length.*

At the top, find the enzyme used to define your tag and double-click to add it to the panel on the right (as it has been done with *NlaIII* in figure 19.125). You can use the filter text box so search for the enzyme name.

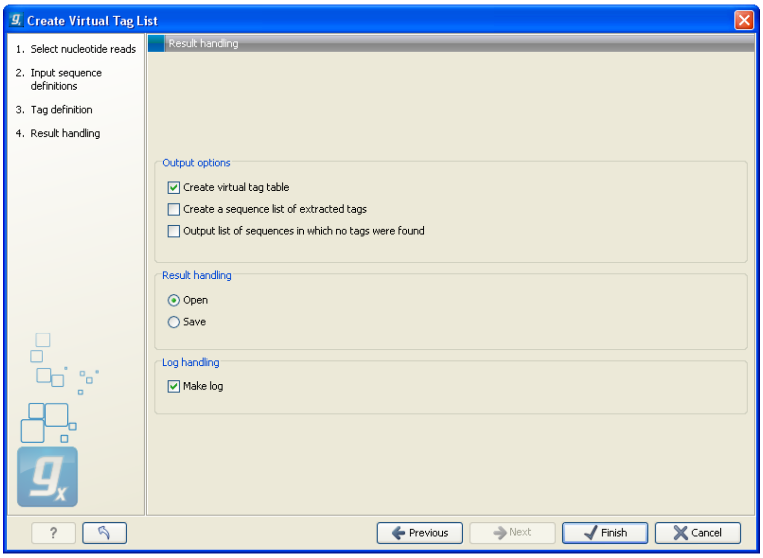
Below, there are further options for the tag extraction:

Extract tags When extracting the virtual tags, you have to decide how to handle the situation where one transcript has several cut sites. In that case there would be several potential tags. Most tag profiling protocols extract the 3'-most tag (as shown in the introduction in figure 19.119), so that would be one way of defining the tags in the virtual tag list. However, due to non-specific cleavage, new alternative splicing or alternative polyadenylation ['t Hoen et al., 2008], tags produced from internal cut sites of the transcript are also quite frequent. This means that it is often not enough to consider the 3'-most restriction site only. The list lets you select either **All**, **External 3'** which is the 3'-most tag or **External 5'** which is the 5' most tag (used by some protocols, for example CAGE - cap analysis of gene expression - see [Maeda et al., 2008]). The result of the analysis displays whether the tag is found at the 3' end or if it is an internal tag (see more below).

Tag downstream/upstream When the cut site is found, you can specify whether the tag is then found downstream or upstream of the site. In figure 19.119, the tag is found downstream.

Tag length The length of the tag to be extracted. This should correspond to the sequence length defined in figure 19.120.

Clicking **Next** allows you to specify the output of the analysis as shown in figure 19.126.



Output options
☑ Create virtual tag table
□ Create a sequence list of extracted tags
□ Output list of sequences in which no tags were found

Result handling
● Open
○ Save

Log handling
☑ Make log

Figure 19.126: *Output options.*

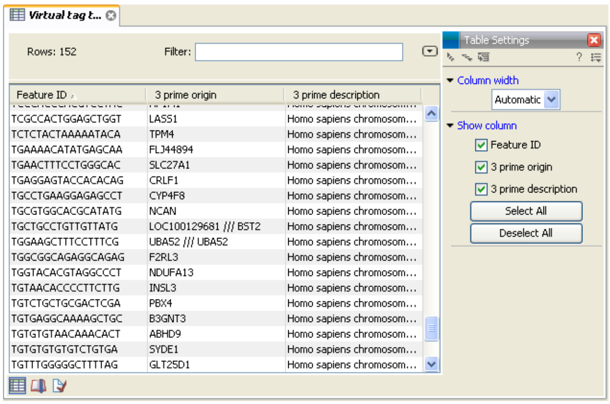
The output options are:

Create virtual tag table This is the primary result listing all the virtual tags. The table is explained in detail below.

Create a sequence list of extracted tags All the extracted tags can be represented in a raw sequence list with no additional information except the name of the transcript. You can e.g. **Export** (📂) this list to a fasta file.

Output list of sequences in which no tags were found The transcripts that do not have a cut site or where the cut site is so close to the end that no tag could be extracted are presented in this list. The list can be used to inspect which transcripts you could potentially fail to measure using this protocol. If there are tags for all transcripts, this list will not be produced.

In figure 19.127 you see an example of a table of virtual tags that have been produced using the **3' external** option described above.



Rows: 152 Filter:

☑ Feature ID
☑ 3 prime origin
☑ 3 prime description

Feature ID	3 prime origin	3 prime description
TCGCCACTGGAGCTGGT	LASS1	Homo sapiens chromosom...
TCTCTACTAAAAATACA	TPM4	Homo sapiens chromosom...
TGAAAACATATGAGCAA	FLJ44894	Homo sapiens chromosom...
TGAACTTTCCTGGGCAC	SLC27A1	Homo sapiens chromosom...
TGAGGAGTACCACACAG	CRLF1	Homo sapiens chromosom...
TGCCTGAAGGAGAGCCT	CYP4F8	Homo sapiens chromosom...
TGCGTGGCACGCATATG	NCAN	Homo sapiens chromosom...
TGCTGCCTGTTGTTATG	LOC100129681 /// BST2	Homo sapiens chromosom...
TGGAAGCTTTCCTTTCG	UBA52 /// UBA52	Homo sapiens chromosom...
TGGCGGCAGAGGCAGAG	F2RL3	Homo sapiens chromosom...
TGGTACACGTAGGCCCT	NDUFA13	Homo sapiens chromosom...
TGTAACACCCCTTCTTG	INSL3	Homo sapiens chromosom...
TGTCTGCTGCGACTCGA	PBX4	Homo sapiens chromosom...
TGTGAGGCAAAAGCTGC	B3GNT3	Homo sapiens chromosom...
TGTGTGTAACAAACACT	ABHD9	Homo sapiens chromosom...
TGTGTGTGTGTCTGTGA	SYDE1	Homo sapiens chromosom...
TGTTTGGGGGCTTTTAG	GLT25D1	Homo sapiens chromosom...

Figure 19.127: *A virtual tag table of 3' external tags.*

The first column lists the tag itself. This is the column used when you annotate your tag count samples or experiments (see section 19.15.3). Next follows the name of the tag's origin transcript. Sometimes the same tag is seen in more than one transcript. In that case, the different origins are separated by /// as it is the case for the tag of *LOC100129681 /// BST2* in figure 19.127. The row just below, *UBA52*, has the same name listed twice. This is because the analysis was based on mRNA annotations from a Refseq genome where each splice variant has its own mRNA annotation, and in this case the *UBA52* gene has two mRNA annotations including the same tag.

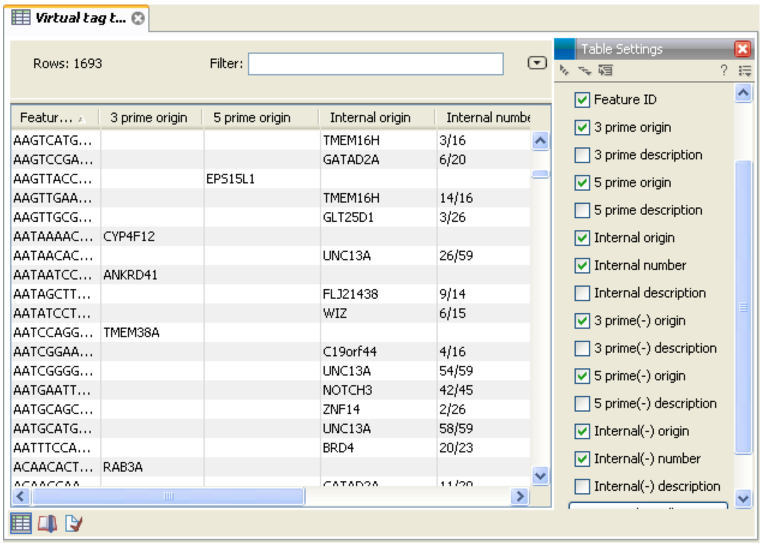
The last column is the description of the transcript (which is either the sequence description if you use a list of un-annotated sequences or all the information in the annotation if you use annotated sequences).

The example shown in figure 19.127 is the simplest case where only the 3' external tags are listed. If you choose to list **All** tags, the table will look like figure 19.128.

In addition to the information about the 3' tags, there are additional columns for 5' and internal tags. For the internal tags there is also a numbering, see for example the top row in figure 19.128 where the *TMEM16H* tag is tag number 3 out of 16. This information can be used to judge how close to the 3' end of the transcript the tag is. As mentioned above, you would often expect to sequence more tags from cut sites near the 3' end of the transcript.

If you have chosen to include reverse complemented sequences in the analysis, there will be an additional set of columns for the tags of the other strand, denoted with a (-).

You can use the advanced table filtering (see section C) to interrogate the number of tags with specific origins (e.g. define a filter where 3' origin != and then leave the text field blank).



Rows: 1693 Filter:

Featur...	3 prime origin	5 prime origin	Internal origin	Internal numbe
AAGTCATG...			TMEM16H	3/16
AAGTCCGA...			GATAD2A	6/20
AAGTTACC...		EPS15L1		
AAGTTGAA...			TMEM16H	14/16
AAGTTGCG...			GLT25D1	3/26
AATAAAAC...	CYP4F12			
AATAACAC...			UNC13A	26/59
AATAATCC...	ANKRD41			
AATAGCTT...			FLJ21438	9/14
AATATCCT...			WIZ	6/15
AATCCAGG...	TMEM38A			
AATCGGAA...			C19orf44	4/16
AATCGGGG...			UNC13A	54/59
AATGAATT...			NOTCH3	42/45
AATGCAGC...			ZNF14	2/26
AATGCATG...			UNC13A	58/59
AATTTCCA...			BRD4	20/23
ACAACACT...	RAB3A			

☑ Feature ID
☑ 3 prime origin
□ 3 prime description
☑ 5 prime origin
□ 5 prime description
☑ Internal origin
☑ Internal number
□ Internal description
☑ 3 prime(-) origin
□ 3 prime(-) description
☑ 5 prime(-) origin
□ 5 prime(-) description
☑ Internal(-) origin
☑ Internal(-) number
□ Internal(-) description

Figure 19.128: *A virtual tag table where all tags have been extracted. Note that some of the columns have been ticked off in the Side Panel.*

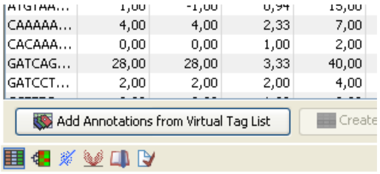
19.15.3 Annotate tag experiment

Combining the tag counts () from the experimental data (see section 19.15.1) with the virtual tag list () (see above) makes it possible to put gene or transcript names on the tag counts. The Workbench simply compares the tags in the experimental data with the virtual tags and transfers the annotations from the virtual tag list to the experimental data.

This is done on an experiment level (experiments are collections of samples with defined groupings, see section 20.1):

> **Toolbox** | **High-throughput Sequencing** () | **Expression Profiling by Tags** () | **Annotate Tag Experiment** ()

You can also access this functionality at the bottom of the **Experiment table** () as shown in figure 19.129.



ATGTAA...	1,00	-1,00	0,94	15,00
CAAAAA...	4,00	4,00	2,33	7,00
CACAAA...	0,00	0,00	1,00	2,00
GATCAG...	28,00	28,00	3,33	40,00
GATCCT...	2,00	2,00	2,00	4,00

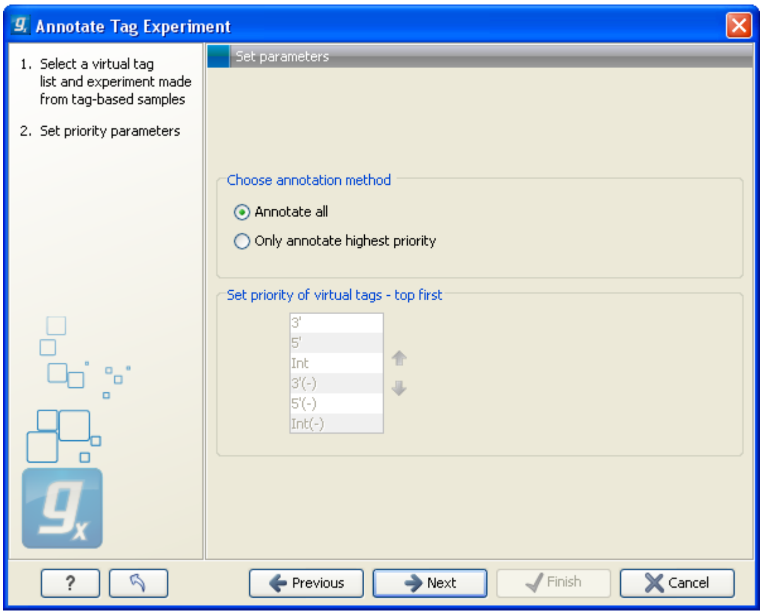
Figure 19.129: *You can annotate an experiment directly from the experiment table.*

This will open a dialog where you select a virtual tag list () and an experiment () of tag-based samples. Click **Next** when the elements are listed in the right-hand side of the dialog.

This dialog lets you choose how you want to annotate your experiment (see figure 19.130).

If a tag in the virtual tag list has more than one origin (as shown in the example in figure 19.128) you can decide how you want your experimental data to be annotated. There are basically two options:

Annotate all This will transfer all annotations from the virtual tag. The type of origin is still preserved so that you can see if it is a 3' external, 5' external or internal tag.



Annotate Tag Experiment

1. Select a virtual tag list and experiment made from tag-based samples
2. Set priority parameters

Set parameters

Choose annotation method

● Annotate all

○ Only annotate highest priority

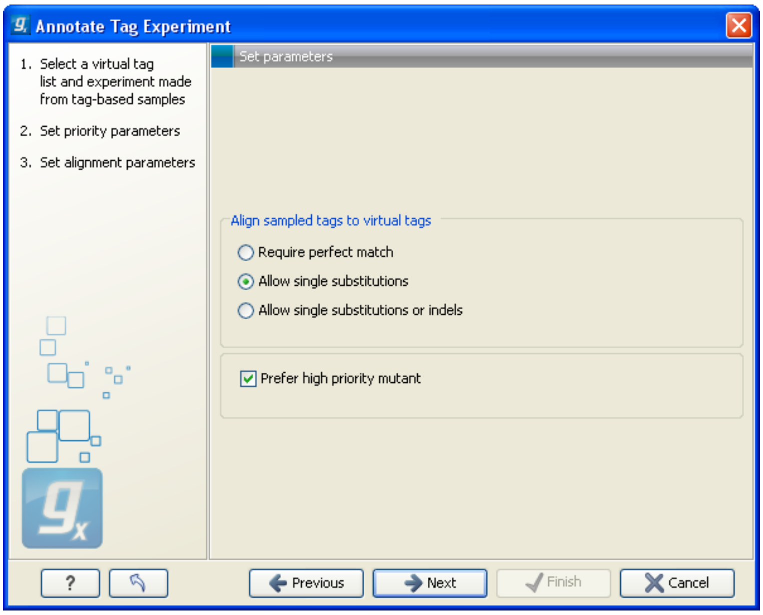
Set priority of virtual tags - top first

3'
5'
Int
3'(-)
5'(-)
Int(-)

Figure 19.130: *Defining the annotation method.*

Only annotate highest priority This will look for the highest priority annotation and only add this to the experiment. This means that if you have a virtual tag with a 3' external and an internal tag, only the 3' external tag will be annotated (using the default prioritization). You can define the prioritization yourself in the table below: simply select a type and press the **up** (⬆) and **down** (⬇) arrows to move it up and down in the list. Note that the priority table is only active when you have selected **Only annotate highest priority**.

Click **Next** to choose how you want to tags to be aligned (see figure 19.131). When the tags from the virtual tag list are compared to your experiment, the tags are matched using one of the following options:



Annotate Tag Experiment

1. Select a virtual tag list and experiment made from tag-based samples
2. Set priority parameters
3. Set alignment parameters

Set parameters

Align sampled tags to virtual tags

○ Require perfect match

● Allow single substitutions

○ Allow single substitutions or indels

☑ Prefer high priority mutant

Figure 19.131: *Settings for aligning the tags.*

```
Tag from experiment:                        CGTATCAATCGATTAC
                                            ||||||||||||||||
Tag1 from virtual tag list (internal):      CGTATCAATCGATTAC
                                            | ||||||||||||||
Tag1 from virtual tag list (3' external):   CCTATCAATCGATTAC
```

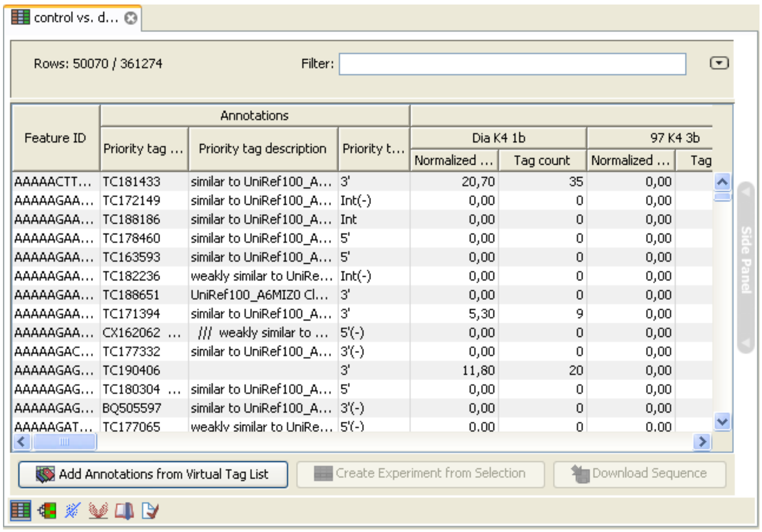
Require perfect match The tags need to be identical to be matched.

Allow single substitutions If there is up to one mismatch in the alignment, the tags will still be matched. If there is a perfect match, single substitutions will not be considered.

Allow single substitutions or indels Similar to the previous option, but now single-base insertions and deletions are also allowed. Perfect matches are preferred to single-base substitutions which are preferred to insertions, which are again preferred to deletions. [3]

If you select any of the two options allowing mismatches or mismatches and indels, you can also choose to **Prefer high priority mutant**. This option is only available if you have chosen to annotate highest priority only in the previous step (see figure 19.130). The option is best explained through an example: In this case, you have a tag that matches perfectly to an internal tag from the virtual tag list. Imagine that in this example, you have prioritized the annotation so that 3' external tags are of higher priority than internal tags. The question is now if you want to accept the perfect match (of a low priority virtual tag) or the high-priority virtual tag with one mismatch? If you check the **Prefer high priority mutant**, the 3' external tag in the example above will be used rather than the perfect match.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. . This will add extra annotation columns to the experiment. The extra columns corresponds to the columns found in your virtual tag list. If you have chosen to annotate highest priority-only, there will only be information from one origin-column for each tag as shown in figure 19.132.


control vs. d...
Filter:
Side Panel
Add Annotations from Virtual Tag List
Create Experiment from Selection
Download Sequence

Rows: 50070 / 361274

Feature ID	Annotations						
				Dia K4 1b		97 K4 3b	
	Priority tag ...	Priority tag description	Priority t...	Normalized ...	Tag count	Normalized ...	Tag
AAAAACTT...	TC181433	similar to UniRef100_A...	3'	20,70	35	0,00	
AAAAAGAA...	TC172149	similar to UniRef100_A...	Int(-)	0,00	0	0,00	
AAAAAGAA...	TC188186	similar to UniRef100_A...	Int	0,00	0	0,00	
AAAAAGAA...	TC178460	similar to UniRef100_A...	5'	0,00	0	0,00	
AAAAAGAA...	TC163593	similar to UniRef100_A...	5'	0,00	0	0,00	
AAAAAGAA...	TC182236	weakly similar to UniRe...	Int(-)	0,00	0	0,00	
AAAAAGAA...	TC188651	UniRef100_A6MIZ0 Cl...	3'	0,00	0	0,00	
AAAAAGAA...	TC171394	similar to UniRef100_A...	3'	5,30	9	0,00	
AAAAAGAA...	CX162062 ...	/// weakly similar to ...	5'(-)	0,00	0	0,00	
AAAAAGAC...	TC177332	similar to UniRef100_A...	3'(-)	0,00	0	0,00	
AAAAAGAG...	TC190406		3'	11,80	20	0,00	
AAAAAGAG...	TC180304 ...	similar to UniRef100_A...	5'	0,00	0	0,00	
AAAAAGAG...	BQ505597	similar to UniRef100_A...	3'(-)	0,00	0	0,00	
AAAAAGAT...	TC177065	weakly similar to UniRe...	5'(-)	0.00	0	0.00	

Figure 19.132: *An experiment annotated with prioritized tags.*

[3] Note that if you use color space data, only color errors are allowed when choosing anything but perfect match.

19.16 Small RNA analysis

The small RNA analysis tools in *CLC Genomics Workbench* are designed to facilitate trimming of sequencing reads, counting and annotating of the resulting tags using miRBase or other annotation sources and performing expression analysis of the results. The tools are general and flexible enough to accommodate a variety of data sets and applications within small RNA profiling, including the counting and annotation of both microRNAs and other non-coding RNAs from any organism. Both Illumina, 454 and SOLiD sequencing platforms are supported. For SOLiD, adapter trimming and annotation is done in color space.

The annotation part is designed to make special use of the information in miRBase but more general references can be used as well.

There are generally two approaches to the analysis of microRNAs or other smallRNAs: (1) count the different types of small RNAs in the data and compare them to databases of microRNAs or other smallRNAs, or (2) map the small RNAs to an annotated reference genome and count the numbers of reads mapped to regions which have smallRNAs annotated. The approach taken by *CLC Genomics Workbench* is (1). This approach has the advantage that it does not require an annotated genome for mapping — you can use the sequences in miRBase or any other sequence list of smallRNAs of interest to annotate the small RNAs. In addition, small RNAs that would not have mapped to the genome (e.g. when lacking a high-quality reference genome or if the RNAs have not been transcribed from the host genome) can still be measured and their expression be compared. The methods and tools developed for *CLC Genomics Workbench* are inspired by the findings and methods described in [Creighton et al., 2009], [Wyman et al., 2009], [Morin et al., 2008] and [Stark et al., 2010].

In the following, the tools for working with small RNAs are described in detail. Look at the tutorials on `http://www.clcbio.com/tutorials` to see examples of analyzing specific data sets.

19.16.1 Extract and count

First step in the analysis is to import the data (see section 19.1).

The next step is to extract and count the small RNAs to create a *small RNA sample* that can be used for further analysis (either annotating or analyzing using the expression analysis tools):

> **Toolbox | High-throughput Sequencing () | Small RNA Analysis () | Extract and Count ()**

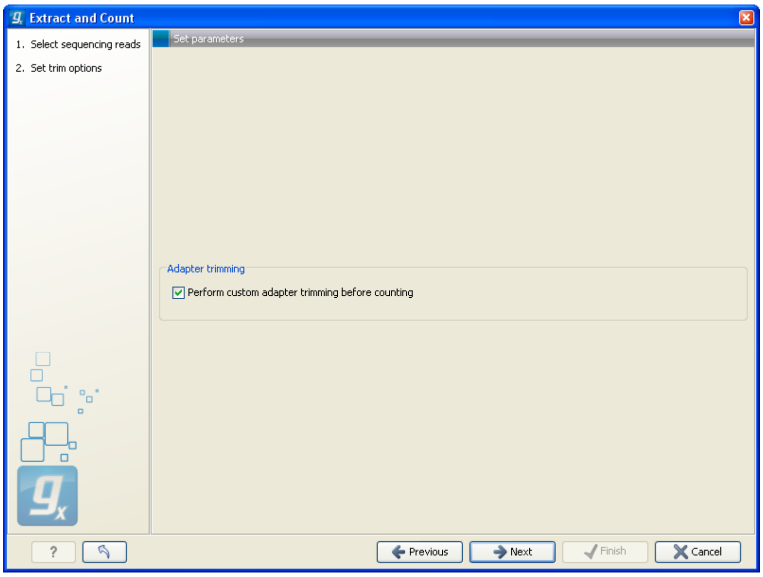
This will open a dialog where you select the sequencing reads that you have imported. Click **Next** when the sequencing data is listed in the right-hand side of the dialog. Note that if you have several samples, they should be processed separately.

This dialog (see figure 19.133) is where you specify whether the reads should be trimmed for adapter sequences prior to counting. It is often necessary to trim off remainders of adapter sequences from the reads before counting.

When you click **Next**, you will be able to specify how the trim should be performed as shown in figure 19.134.

If you have chosen not to trim the reads for adapter sequence, you will see figure 19.135 instead.

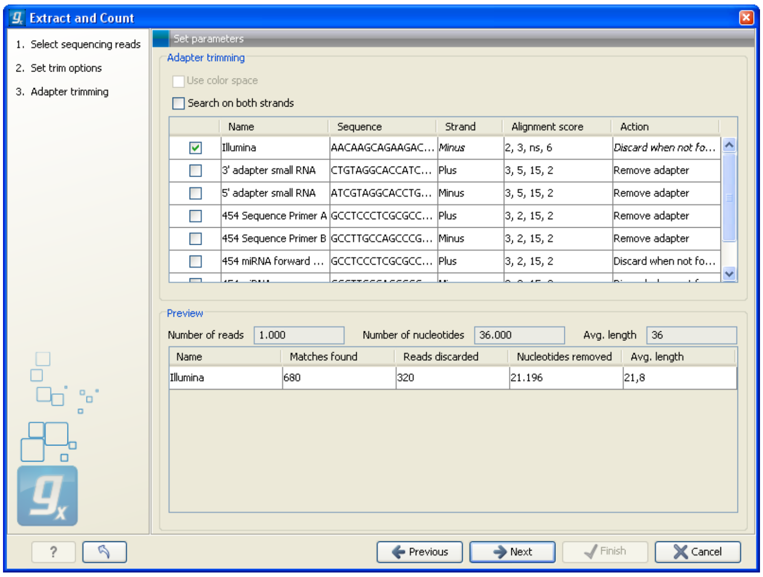
The trim options shown in figure 19.134 are the same as described under adapter trim in section 19.3.2. Please refer to this section for more information.



Adapter trimming

☑ Perform custom adapter trimming before counting

Figure 19.133: *Specifying whether adapter trimming is needed.*



Adapter trimming

□ Use color space

□ Search on both strands

	Name	Sequence	Strand	Alignment score	Action
☑	Illumina	AACAAGCAGAAGAC...	*Minus*	2, 3, ns, 6	*Discard when not fo...*
□	3' adapter small RNA	CTGTAGGCACCATC...	Plus	3, 5, 15, 2	Remove adapter
□	5' adapter small RNA	ATCGTAGGCACCTG...	Minus	3, 5, 15, 2	Remove adapter
□	454 Sequence Primer A	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Remove adapter
□	454 Sequence Primer B	GCCTTGCCAGCCCG...	Minus	3, 2, 15, 2	Remove adapter
□	454 miRNA forward ...	GCCTCCCTCGCGCC...	Plus	3, 2, 15, 2	Discard when not fo...

Preview

Number of reads 1.000 Number of nucleotides 36.000 Avg. length 36

Name	Matches found	Reads discarded	Nucleotides removed	Avg. length
Illumina	680	320	21.196	21,8

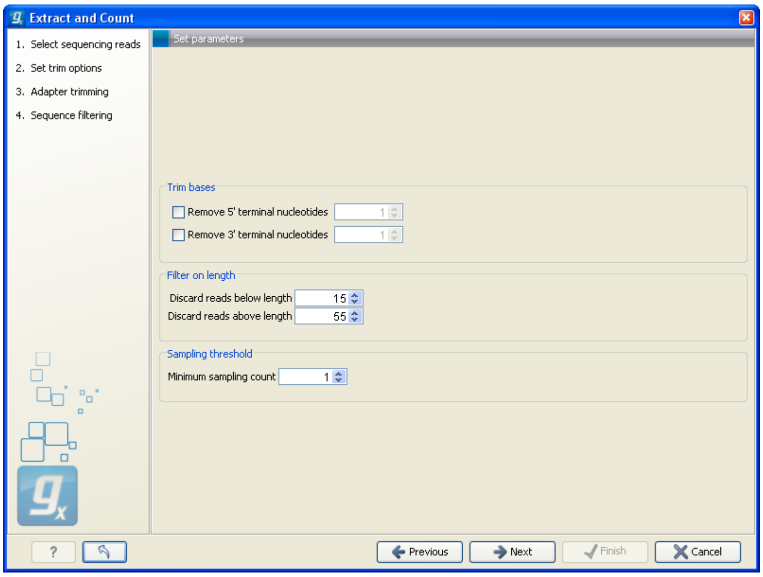
Figure 19.134: *Setting parameters for adapter trim.*

It should be noted that if you expect to see part of adapters in your reads, you would typically choose **Discard when not found** as the action. By doing this, only reads containing the adapter sequence will be counted as small RNAs in the further analysis. If you have a data set where the adapter may be there or not you would choose **Remove adapter**.

Note that all reads will be trimmed for ambiguity symbols such as N before the adapter trim.

Clicking **Next** allows you to specify additional options regarding trimming and counting as shown in figure 19.135.

At the top you can choose to **Trim bases** by specifying a number of bases to be removed from either the 3' or the 5' end of the reads. Below, you can specify the minimum and maximum lengths of the small RNAs to be counted (this is the length after trimming). The minimum length that can be set is 15 and the maximum is 55.



Trim bases

- ☐ Remove 5' terminal nucleotides 1
- ☐ Remove 3' terminal nucleotides 1

Filter on length

- Discard reads below length 15
- Discard reads above length 55

Sampling threshold

- Minimum sampling count 1

Figure 19.135: *Defining length interval and sampling threshold.*

At the bottom, you can specify the **Minimum sampling count**. This is the number of copies of the small RNAs (tags) that are needed in order to include it in the resulting count table (the small RNA sample). The actual counting is very simple and relies on **perfect match** between the reads to be counted together[4]. This also means that a count threshold of 1 will include a lot of unique tags as a result of sequencing errors. In order to set the threshold right, the following should be considered:

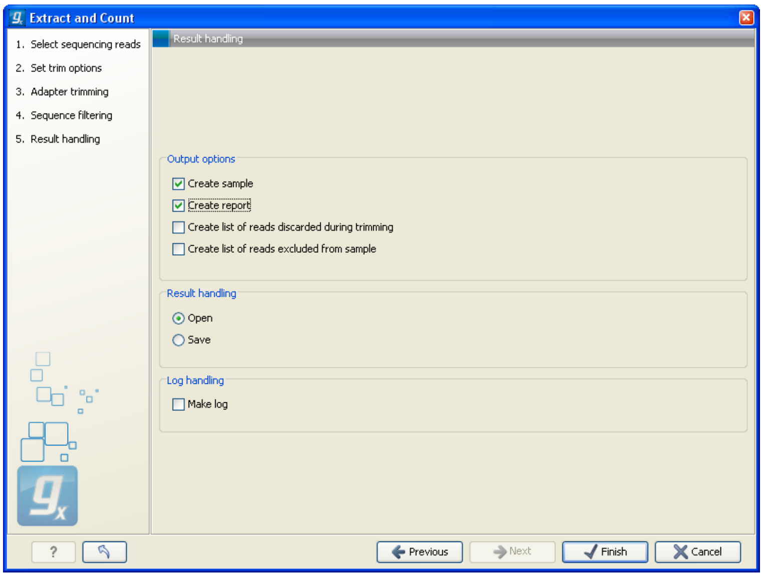
- If the sample is going to *be annotated*, annotations may be found for the tags resulting from sequencing errors. This means that there is no negative effect of including tags with a low count in the output.
- When using *un-annotated sequences* for discovery of novel small RNAs, it may be useful to apply a higher threshold to eliminate the noise from sequencing errors. However, this can be done at a later stage by filtering the sample and creating a sub-set.
- When multiple samples are compared, it is interesting to know if one tag which is abundant in one sample is also found in another, even at a very low number. In this case, it is useful to include the tags with very low counts, since they may become more trustworthy in combination with information from other samples.
- Setting the count threshold higher will reduce the size of the sample produced which will reduce the memory and disk usage when working with the results.

Clicking **Next** allows you to specify the output of the analysis as shown in 19.136.

The options are:

Create sample This is the primary result showing all the tags and respective counts (an example is shown in figure 19.137). Each row represents a tag with the actual sequence as the feature ID and a column with **Length** and **Count**. The actual count is based on 100 %

[4] Note that you can identify variants of the same miRNA when annotating the sample (see below).



Extract and Count

1. Select sequencing reads
2. Set trim options
3. Adapter trimming
4. Sequence filtering
5. Result handling

Result handling

Output options

☑ Create sample

☑ Create report

□ Create list of reads discarded during trimming

□ Create list of reads excluded from sample

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 19.136: *Output options.*

similarity[5]. The sample can be used in further analysis by the expression analysis tools (see chapter 20) in the "raw" form, or you can annotate it (see below). The tools for working with the data in the sample are described in section 19.16.4.

Create report This will create a summary report as described below.

Create list of reads discarded during trimming This list contains the reads where no adapter was found (when choosing **Discard when not found** as the action).

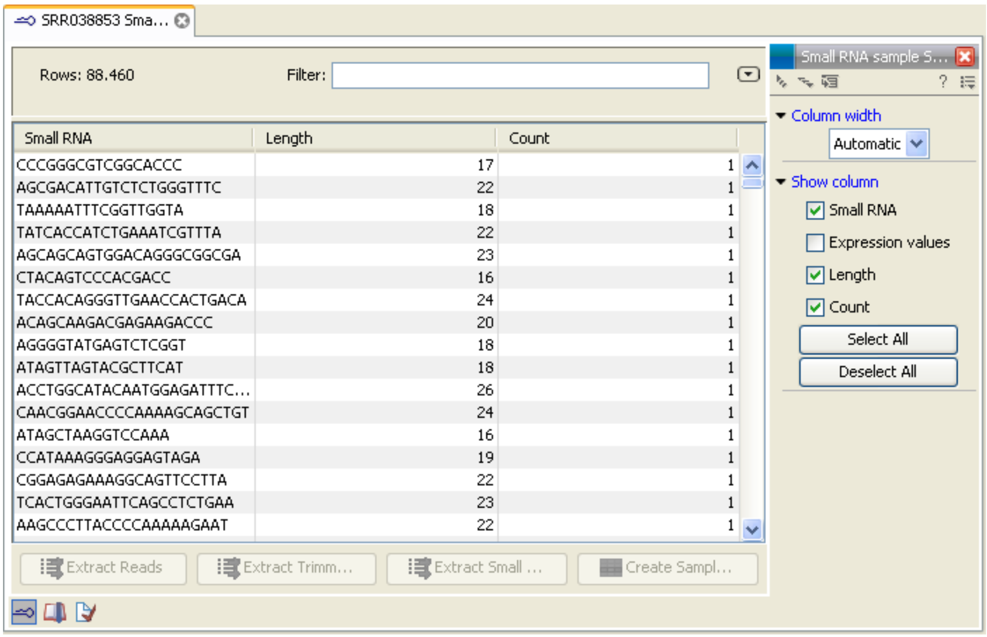
Create list of reads excluded from sample This list contains the reads that passed the trimming but failed to meet the sampling thresholds regarding minimum/maximum length and number of copies.

The summary report includes the following information (an example is shown in figure 19.138):

Trim summary Shows the following information for each input file:

- Number of reads in the input.
- Average length of the reads in the input.
- Number of reads after trim. The difference between the number of reads in the input and this number will be the number of reads that are discarded by the trim.
- Percentage of the reads that pass the trim.
- Average length after trim. When analyzing miRNAs, you would expect this number to be around 22. If the number is significantly lower or higher, it could indicate that the trim settings are not right. In this case, check that the trim sequence is correct, that the strand is right, and adjust the alignment scores. Sometimes it is preferable to increase the minimum scores to get rid of low-quality reads. The average length after trim could also be somewhat larger than 22 if your sequenced data contains a mixture of miRNA and other (longer) small RNAs.

[5] Note that you can identify variants of the same miRNA when annotating the sample (see below).



Rows: 88.460 Filter:

Small RNA	Length	Count
CCCGGGCGTCGGCACCC	17	1
AGCGACATTGTCTCTGGGTTTC	22	1
TAAAAATTTCGGTTGGTA	18	1
TATCACCATCTGAAATCGTTTA	22	1
AGCAGCAGTGGACAGGGCGGCGA	23	1
CTACAGTCCCACGACC	16	1
TACCACAGGGTTGAACCACTGACA	24	1
ACAGCAAGACGAGAAGACCC	20	1
AGGGGTATGAGTCTCGGT	18	1
ATAGTTAGTACGCTTCAT	18	1
ACCTGGCATACAATGGAGATTTC...	26	1
CAACGGAACCCCAAAAGCAGCTGT	24	1
ATAGCTAAGGTCCAAA	16	1
CCATAAAGGGAGGAGTAGA	19	1
CGGAGAGAAAGGCAGTTCCTTA	22	1
TCACTGGGAATTCAGCCTCTGAA	23	1
AAGCCCTTACCCCAAAAAGAAT	22	1

Column width: Automatic

Show column

☑ Small RNA

☐ Expression values

☑ Length

☑ Count

Figure 19.137: *The tags have been extracted and counted.*

Read length before/after trimming Shows the distribution of read lengths before and after trim. The graph shown in figure 19.138 is typical for miRNA sequencing where the read lengths after trim peaks at 22 bp.

Trim settings The trim settings summarized. Note that ambiguity characters will automatically be trimmed.

Detailed trim results This is described under adapter trim in section 19.3.2.

Tag counts The number of tags and two plots showing on the x-axis the counts of tags and on the y-axis the number of tags for which this particular count is observed. The plot is in a zoomed version where only the lower part of the y-axis is shown to make it possible to see the numbers of tags higher counts.

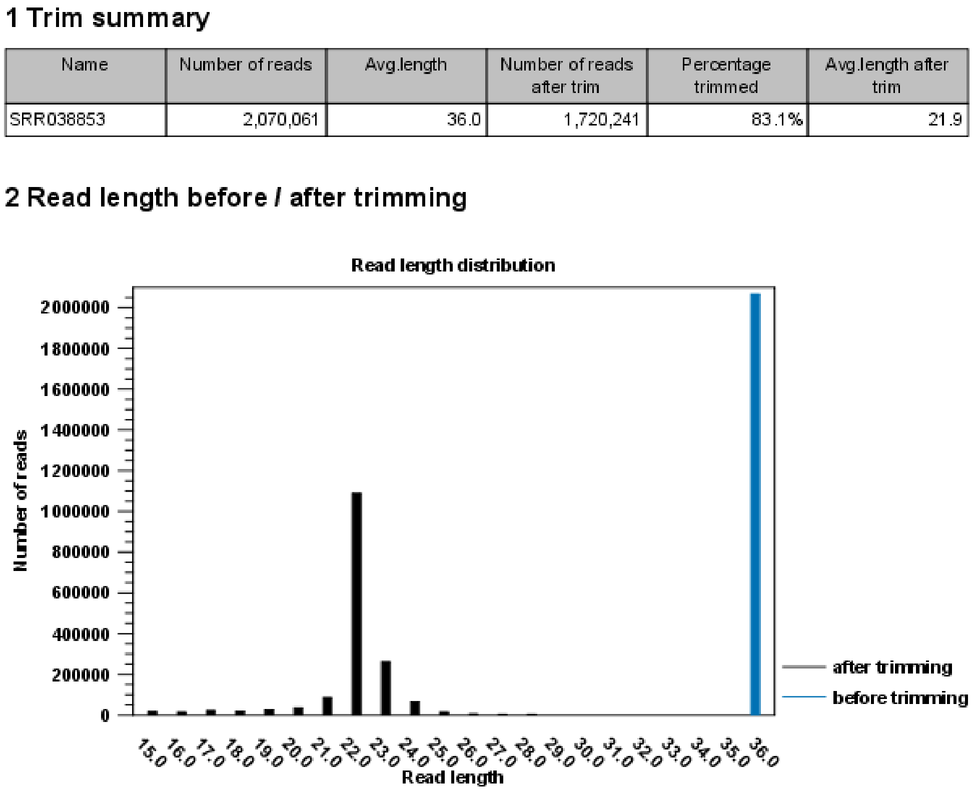
19.16.2 Downloading miRBase

In order to make use of the additional information about mature and mature* regions on the precursor miRNAs in miRBase, you need to use the integrated tool to download miRBase rather than downloading it from http://www.mirbase.org/:

Toolbox | High-throughput Sequencing ( ) | Small RNA Analysis ( ) | Download miRBase ( )

This will download a sequence list with all the precursor miRNAs including annotations for mature and mature* regions. The list can then be selected when annotating the samples with miRBase (see section 19.16.3).

The downloaded version will always be the latest version (it is downloaded from ftp://mirbase.org/pub/mirbase/CURRENT/miRNA.dat.gz). Information on the version number of miRBase is also available in the **History** ( ) of the downloaded sequence list, and when using this for annotation, the annotated samples will also include this information in their **History** ( ).



1 Trim summary

Name	Number of reads	Avg.length	Number of reads after trim	Percentage trimmed	Avg.length after trim
SRR038853	2,070,061	36.0	1,720,241	83.1%	21.9

2 Read length before / after trimming

Figure 19.138: *A summary report of the counting.*

19.16.3 Annotating and merging small RNA samples

The small RNA sample produced when counting the tags (see section 19.16.1) can be enriched by *CLC Genomics Workbench* by comparing the tag sequences with annotation resources such as miRBase and other small RNA annotation sources. Note that the annotation can also be performed on an experiment, set up from small RNA samples (see section 20.1.2).

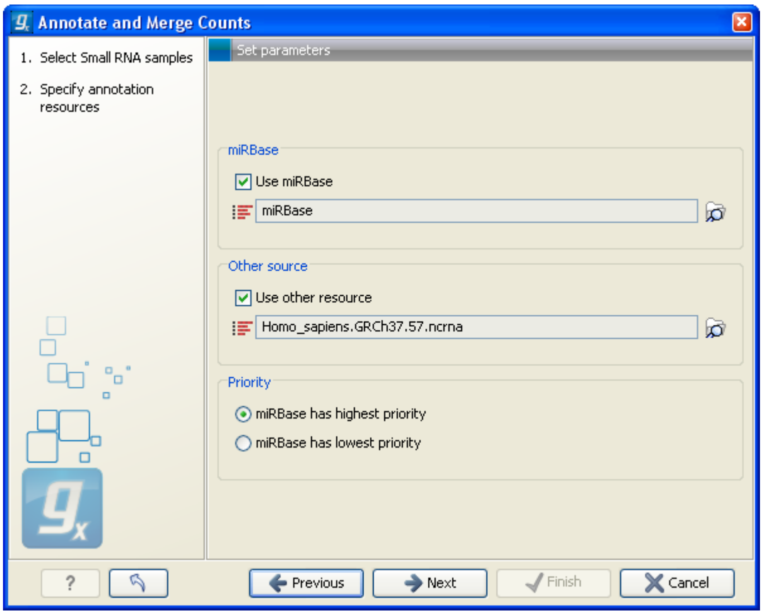
Besides adding annotations to known small RNAs in the sample, it is also possible to merge variants of the same small RNA to get a cumulated count. When initially counting the tags, the Workbench requires a that the trimmed reads are identical for them to be counted as the same tag. However, you will often see different variants of the same miRNA in a sample, and it is useful to be able to count these together. This is also possible using the tool to annotate and merge samples.

> **Toolbox** | **High-throughput Sequencing ()** | **Small RNA Analysis ()** | **Annotate and Merge Counts ()**

This will open a dialog where you select the small RNA samples () to be annotated. Note that if you have included several samples, they will be processed separately but summarized in one report providing a good overview of all samples. You can also input **Experiments** () (see section 20.1.2) created from small RNA samples. Click **Next** when the data is listed in the right-hand side of the dialog.

This dialog (figure 19.139) is where you define the annotation resources to be used.

There are two ways of providing annotation sources:



Annotate and Merge Counts

1. Select Small RNA samples
2. Specify annotation resources

Set parameters

miRBase

☑ Use miRBase

miRBase

Other source

☑ Use other resource

Homo_sapiens.GRCh37.57.ncrna

Priority

● miRBase has highest priority

○ miRBase has lowest priority

Figure 19.139: *Defining annotation resources.*

- Downloading miRBase using the integrated download tool (explained in section 19.16.2).
- Importing a list of sequences, e.g. from a fasta file. This could be from Ensembl, e.g. ftp://ftp.ensembl.org/pub/release-57/fasta/homo_sapiens/ncrna/Homo_sapiens.GRCh37.57.ncrna.fa.gz or from ncRNA.org: http://www.ncrna.org/frnadb/files/ncrna.zip.

The downloaded miRBase file contains all precursor sequences from the latest version of miRBase http://www.mirbase.org/ including annotations defining the mature and optionally mature* regions (see an example in figure 19.140).

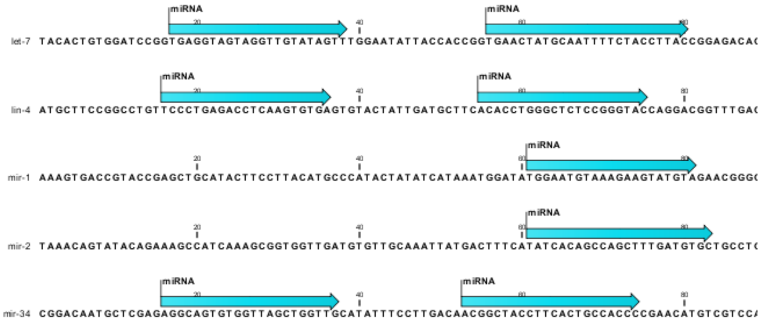
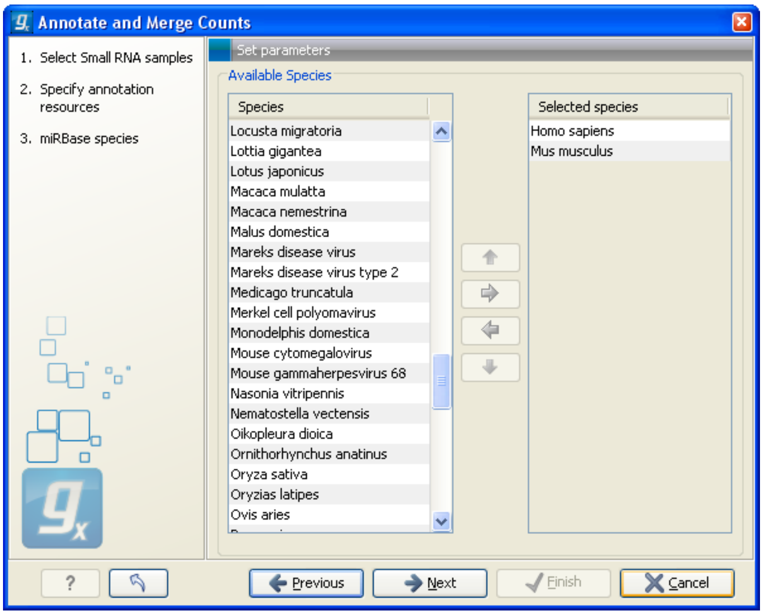

miRNA
miRNA
let-7 TACACTGTGGATCCGGTGAGGTAGTAGGTTGTATAGTTTGGAATATTACCACCGGTGAACTATGCAATTTTCTACCTTACCGGAGACA
miRNA
miRNA
lin-4 ATGCTTCCGGCCTGTTCCCTGAGACCTCAAGTGTGAGTGTACTATTGATGCTTCACACCTGGGCTCTCCGGGTACCAGGACGGTTTGA
miRNA
mir-1 AAAGTGACCGTACCGAGCTGCATACTTCCTTACATGCCCATACTATATCATAAATGGATATGGAATGTAAAGAAGTATGTAGAACGGG
miRNA
mir-2 TAAACAGTATACAGAAAGCCATCAAAGCGGTGGTTGATGTGTTGCAAATTATGACTTTCATATCACAGCCAGCTTTGATGTGCTGCCT
miRNA
miRNA
mir-34 CGGACAATGCTCGAGAGGCAGTGTGGTTAGCTGGTTGCATATTTCCTTGACAACGGCTACCTTCACTGCCACCCCGAACATGTCGTCC
20
40
60
80

Figure 19.140: *Some of the precursor miRNAs from miRBase have both mature and mature* regions annotated (as the two first in this list).*

This means that it is possible to have a more fine-grained classification of the tags using miRBase compared to a simple fasta file resource containing the full precursor sequence. This is the reason why the miRBase annotation source is specified separately in figure 19.139.

At the bottom of the dialog, you can specify whether miRBase should be prioritized over the additional annotation resource. The prioritization is explained in detail later in this section. To prioritize one over the other can be useful when there is redundant information (e.g. if you have an additional source that also contains all the miRNAs from miRBase and you prefer the miRBase annotations when possible).

When you click **Next**, you will be able to choose which species from miRBase that should be used and in which order (see figure 19.141). Note that if you have not selected a miRBase annotation source, you will go directly to the next step shown in figure 19.142.



1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species

Species
Locusta migratoria
Lottia gigantea
Lotus japonicus
Macaca mulatta
Macaca nemestrina
Malus domestica
Mareks disease virus
Mareks disease virus type 2
Medicago truncatula
Merkel cell polyomavirus
Monodelphis domestica
Mouse cytomegalovirus
Mouse gammaherpesvirus 68
Nasonia vitripennis
Nematostella vectensis
Oikopleura dioica
Ornithorhynchus anatinus
Oryza sativa
Oryzias latipes
Ovis aries

Selected species
Homo sapiens
Mus musculus

Figure 19.141: *Defining and prioritizing species in miRBase.*

To the left, you see the list of species in miRBase. This list is dynamically created based on the information in the miRBase file. Using the arrow button (➡) you can add species to the right-hand panel. Note that the order of the species is important since the tags are annotated iteratively based on the order specified here. This means that in the example in figure 19.141, a human miRNA will preferred over mouse, even if they are identical in sequence (the prioritization is elaborated below). The up and down arrows (⬆)/ (⬇) can be used to change the order of species.

When you click **Next**, you will be able to specify how the alignment of the tags against the annotation sources should be performed (see figure 19.142).

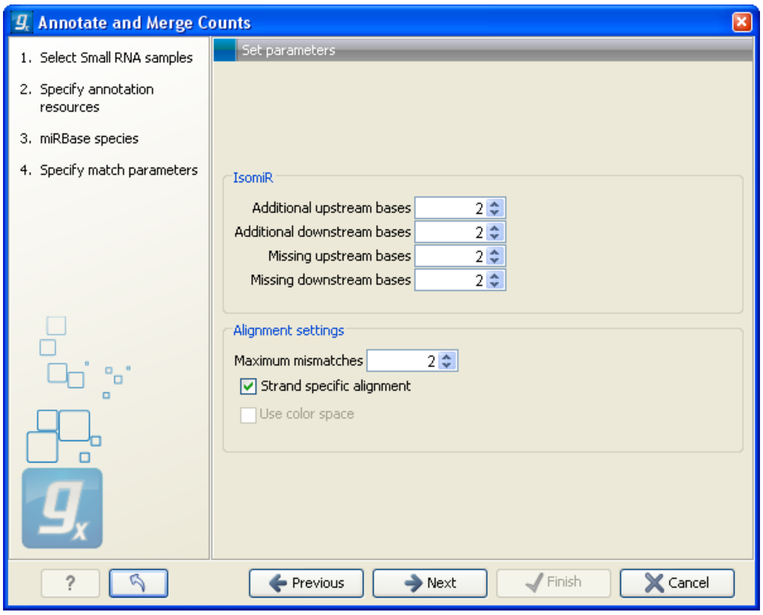
The panel at the top is active only if you have chosen to annotate with miRBase. It is used to define the requirements to the alignment of a read for it to be counted as a mature or mature* tag:

Additional upstream bases This defines how many bases the tag is allowed to extend the annotated mature region at the 5' end and still be categorized as mature.

Additional downstream bases This defines how many bases the tag is allowed to extend the annotated mature region at the 3' end and still be categorized as mature.

Missing upstream bases This defines how many bases the tag is allowed to miss at the 5' end compared to the annotated mature region and still be categorized as mature.

Missing downstream bases This defines how many bases the tag is allowed to miss at the 3' end compared to the annotated mature region and still be categorized as mature.


Annotate and Merge Counts
1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species
4. Specify match parameters
Set parameters
?
Previous
Next
Finish
Cancel

IsomiR

Additional upstream bases 2

Additional downstream bases 2

Missing upstream bases 2

Missing downstream bases 2

Alignment settings

Maximum mismatches 2

☑ Strand specific alignment

□ Use color space

Figure 19.142: *Setting parameters for aligning.*

At the bottom of the dialog you can specify the **Maximum mismatches** (default value is 2). Furthermore, you can specify if the alignment and annotation should be performed in **color space** which is available when your small RNA sample is based on SOLiD data. [6]

At this point, a more elaborate explanation of the annotation algorithm is needed. The short read mapping algorithm in the *CLC Genomics Workbench* is used to map all the tags to the reference sequences which comprise the full precursor sequences from miRBase and the sequence lists chosen as additional resources. The mapping is done in several rounds: the first round is done requiring a perfect match, the second allowing one mismatch, the third allowing two mismatches etc. No gaps are allowed. The number of rounds depend on the number of mismatches allowed[7] (default is two which means three rounds of read mapping, see figure 19.142).

After each round of mapping, the tags that are mapped will be removed from the list of tags that continue to the next round. This means that a tag mapping with perfect match in the first round will not be considered for the subsequent one-mismatch round of mapping.

Following the mapping, the tags are classified into the following categories according to where they match:

- Mature exact
- Mature super
- Mature sub
- Mature sub/super
- Mature* exact

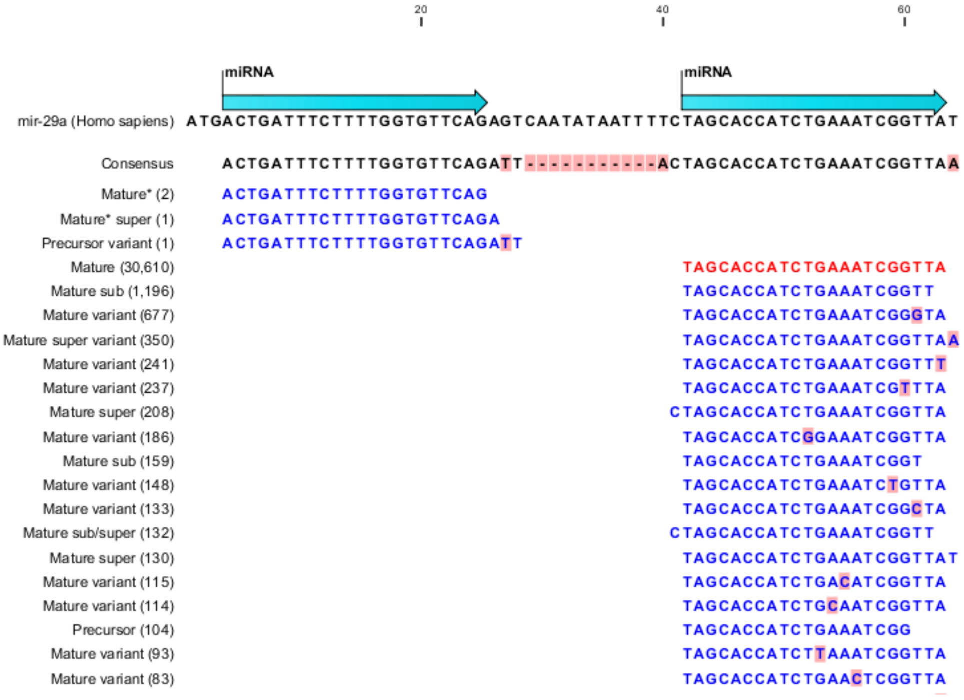
[6] Note that this option is only going to make a difference for tags with low counts. Since the actual tag counting in the first place is done based on perfect matches, the highly abundant tags are not likely to have sequencing errors, and aligning in color space does not add extra benefit for these.

[7] For color space, the maximum number of mismatches is 2.

- Mature* super
- Mature* sub
- Mature* sub/super
- Precursor
- Other

All these categories except *Other* refer to hits in miRBase. The *Other* category is for hits in the other resources (the information about resource is also shown in the output). *Sub* means that the observed tag is shorter than the annotated mature or mature*; *super* means that the observed tag is longer than the annotated mature or mature*. The combination *sub/super* means that the observed tag extends the annotation in one end and is shorter at the other end.

An example of an alignment is shown in figure 19.143 using the same alignment settings as in figure 19.142.



```
mir-29a (Homo sapiens) ATGACTGATTTCTTTTGGTGTTCAGAGTCAATATAATTTTCTAGCACCATCTGAAATCGGTTAT

             Consensus    ACTGATTTCTTTTGGTGTTCAGATT-----------ACTAGCACCATCTGAAATCGGTTAA
           Mature* (2)    ACTGATTTCTTTTGGTGTTCAG
      Mature* super (1)   ACTGATTTCTTTTGGTGTTCAGA
  Precursor variant (1)   ACTGATTTCTTTTGGTGTTCAGATT
      Mature (30,610)                                          TAGCACCATCTGAAATCGGTTA
     Mature sub (1,196)                                        TAGCACCATCTGAAATCGGTT
    Mature variant (677)                                       TAGCACCATCTGAAATCGGGTA
Mature super variant (350)                                     TAGCACCATCTGAAATCGGTTAA
    Mature variant (241)                                       TAGCACCATCTGAAATCGGTTT
    Mature variant (237)                                       TAGCACCATCTGAAATCGTTTA
      Mature super (208)                                      CTAGCACCATCTGAAATCGGTTA
    Mature variant (186)                                       TAGCACCATCGGAAATCGGTTA
        Mature sub (159)                                       TAGCACCATCTGAAATCGGT
    Mature variant (148)                                       TAGCACCATCTGAAATCTGTTA
    Mature variant (133)                                       TAGCACCATCTGAAATCGGCTA
  Mature sub/super (132)                                      CTAGCACCATCTGAAATCGGTT
      Mature super (130)                                       TAGCACCATCTGAAATCGGTTAT
    Mature variant (115)                                       TAGCACCATCTGACATCGGTTA
    Mature variant (114)                                       TAGCACCATCTGCAATCGGTTA
         Precursor (104)                                       TAGCACCATCTGAAATCGG
     Mature variant (93)                                       TAGCACCATCTTAAATCGGTTA
     Mature variant (83)                                       TAGCACCATCTGAACTCGGTTA
```

Figure 19.143: *Alignment of length variants of mir-30a.*

The two tags at the top are both classified as *mature super* because they cover and extend beyond the annotated mature RNA. The third tag is identical to the annotated mature. The fourth tag is classified as *precursor* because it does not meet the requirements on length for it to be counted as a mature hit — it lacks 6 bp compared to the annotated mature RNA. The fifth tag is classified as mature sub because it also lacks one base but stays within the threshold defined in figure 19.142.

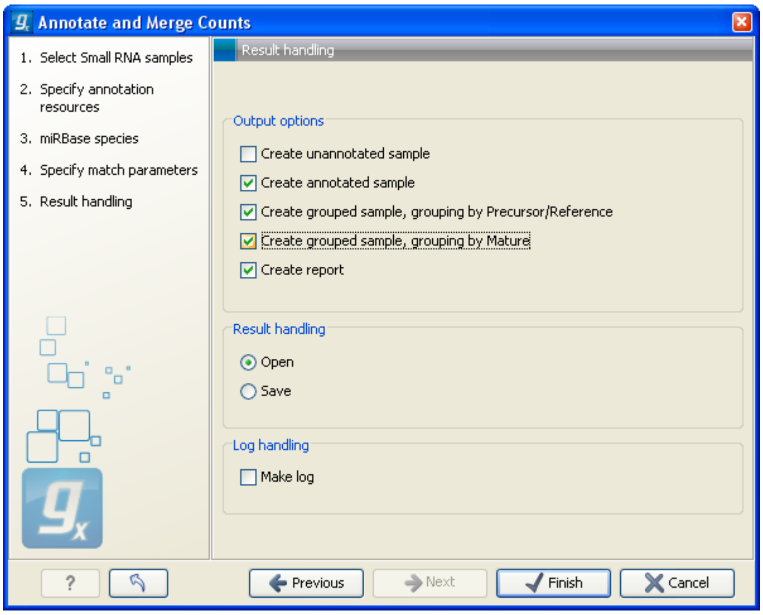
If a tag has several hits, the list above is used for prioritization. This means that e.g. a *Mature sub* is preferred over a *Mature* exact*. Note that if miRBase was chosen as lowest priority (figure

19.139), the *Other* category will be at the top of the list. All tags mapping to a miRBase reference without qualifying to any of the mature and mature* types will be typed as *Precursor*.

In case you have selected more than one species for miRBase annotation (e.g. Homo Sapiens and Mus Musculus) the following rules for adding annotations apply:

1. If a tag has hits with the same priority for both species, the annotation for the top-prioritized species will be added.
2. Read category priority is stronger than species category priority: If a read is a higher priority match for a mouse miRBase sequence than it is for a human miRBase sequence the annotation for the mouse will be used

Clicking **Next** allows you to specify the output of the analysis as shown in 19.144.


Annotate and Merge Counts
1. Select Small RNA samples
2. Specify annotation resources
3. miRBase species
4. Specify match parameters
5. Result handling
Result handling
?
Previous
Next
Finish
Cancel

Output options

- ☐ Create unannotated sample
- ☑ Create annotated sample
- ☑ Create grouped sample, grouping by Precursor/Reference
- ☑ Create grouped sample, grouping by Mature
- ☑ Create report

Result handling

- ● Open
- ○ Save

Log handling

- ☐ Make log

Figure 19.144: *Output options.*

The options are:

Create unannotated sample All the tags where no hit was found in the annotation source are included in the unannotated sample. This sample can be used for investigating novel miRNAs, see section 19.16.5. No extra information is added, so this is just a subset of the input sample.

Create annotated sample This will create a sample as described in section 19.16.4. In this sample, the following columns have been added to the counts.

Name This is the name of the annotation sequence in the annotation source. For miRBase, it will be the names of the miRNAs (e.g. *let-7g* or *mir-147*), and for other source, it will be the name of the sequence.

Resource This is the source of the annotation, either miRBase (in which case the species name will be shown) or other sources (e.g. Homo_sapiens.GRCh37.57.ncrna).

Match type The match type can be exact or variant (with mismatches) of the following types:

- Mature
- Mature super
- Mature sub
- Mature sub/super
- Mature*
- Mature* super
- Mature* sub
- Mature* sub/super
- Precursor
- Other

Mismatches The number of mismatches.

Note that if a tag has two equally prioritized hits, they will be shown with // between the names. This could be e.g. two precursor sequences sharing the same mature sequence (also see the sample grouped on mature below).

Create grouped sample, grouping by Precursor/Reference This will create a sample as described in section 19.16.4. All variants of the same reference sequence will be merged to create one expression value for all.

Expression values. The expression value can be changed at the bottom of the table. The default is to use the counts in the mature column.

Name. The name of the reference. For miRBase this will then be the name of the precursor.

Resource. The name of the resource that the reference comes from.

Exact mature. The number of exact mature reads.

Mature. The number of all mature reads including sub, super and variants.

Unique exact mature. In cases where one tag has several hits (as denoted by the // in the ungrouped annotated sample as described above), the counts are distributed evenly across the references. The difference between *Exact mature* and *Unique exact mature* is that the latter only includes reads that are unique to this reference.

Unique mature. Same as above but for all matures including sub, super and variants.

Exact mature*. Same as above, but for mature*.

Mature*. Same as above, but for mature*.

Unique exact mature*. Same as above, but for mature*.

Unique mature*. Same as above, but for mature*.

Exact other. Exact match in the resources chosen besides miRBase.

Other. All matches in the resources chosen besides miRBase including variants. The last two numbers are the only ones used when the reference is not from miRBase.

Total. The total number of tags mapped and classified to the precursor/reference sequence.

Create grouped sample, grouping by Mature This will create a sample as described in section 19.16.4. This is also a grouped sample, but in addition to grouping based on the same reference sequence, the tags in this sample are grouped on the same mature. This means that two precursor variants of the same mature miRNA are merged. Note that it is only possible to create this sample when using miRBase as annotation resource (because the Workbench has a special interpretation of the miRBase annotations for mature as described previously). To find identical mature miRNAs, the Workbench compares all the mature sequences and when they are identical, they are merged. The names of the precursor sequences merged are all shown in the table.

Expression values. The expression value can be changed at the bottom of the table. The default is to use the counts in the mature column.

Name. The name of the reference. When several precursor sequences have been merged, all the names will be shown separated by //.

Resource. The species of the reference.

Exact mature. The number of exact mature reads.

Mature. The number of all mature reads including sub, super and variants.

Unique exact mature. In cases where one tag has several hits (as denoted by the // in the ungrouped annotated sample as described above), the counts are distributed evenly across the references. The difference between *Exact mature* and *Unique exact mature* is that the latter only includes reads that are unique to one of the precursor sequences that are represented under this mature sequence.

Unique mature. Same as above but for all matures including sub, super and variants.

Create report. A summary report described below.

The summary report includes the following information (an example is shown in figure 19.145):

Summary Shows the following information for each input sample:

- Number of small RNAs(tags) in the input.
- Number of annotated tags (number and percentage).
- Number of reads in the sample (one tag can represent several reads)
- Number of annotated reads (number and percentage).

Resources Shows how many matches were found in each resource:

- Number of sequences in the resource.
- Number of sequences where a match was found (i.e. this sequence has been observed at least once in the sequencing data).

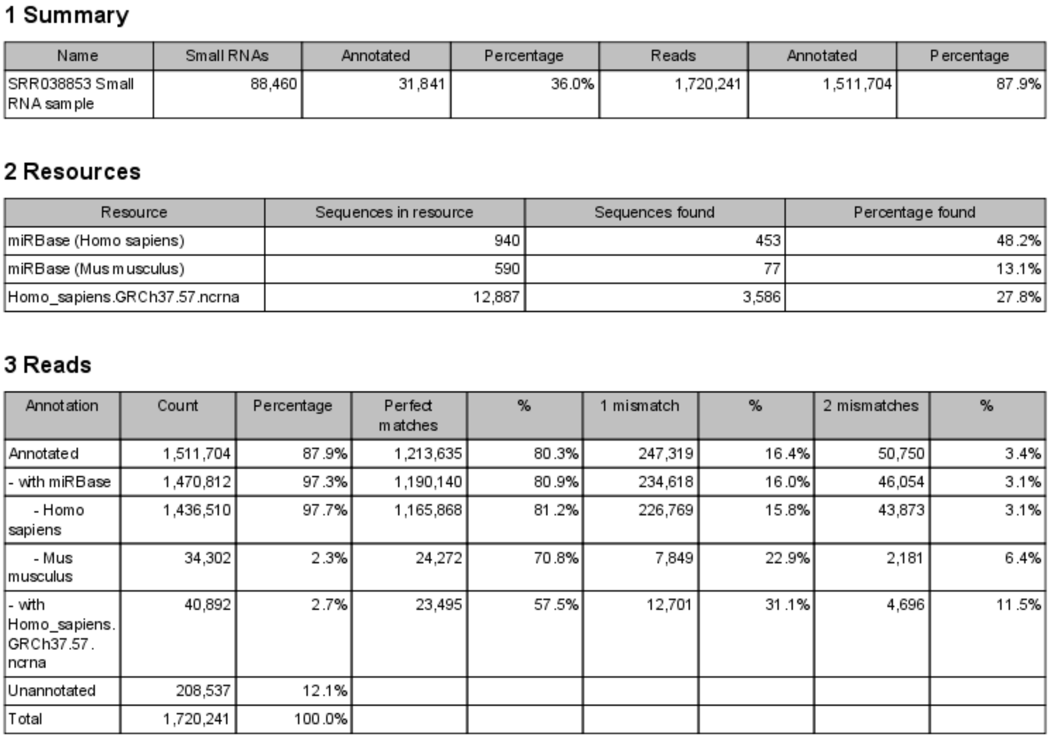
Reads Shows the number of reads that fall into different categories (there is one table per input sample). On the left hand side are the annotation resources. For each resource, the count and percentage of reads in that category are shown. Note that the percentage are relative to the overall categories (e.g. the miRBase reads are a percentage of all the *annotated* reads, not all reads). This is information is shown for each mismatch level.

Small RNAs Similar numbers as for the reads but this time for each small RNA tag and without mismatch differentiation.

Read count proportions A histogram showing, for each interval of read counts, the proportion of annotated (respectively, unannotated) small RNAs with a read count in that interval. Annotated small RNAs may be expected to be associated with higher counts, since the most abundant small RNAs are likely to be known already.

Annotations (miRBase) Shows an overview table for classifications of the number of reads that fall in the miRBase categories for each species selected.

Annotations (Other) Shows an overview table with read numbers for total, exact match and mutant variants for each of the other annotation resources.



1 Summary

Name	Small RNAs	Annotated	Percentage	Reads	Annotated	Percentage
SRR038853 Small RNA sample	88,460	31,841	36.0%	1,720,241	1,511,704	87.9%

2 Resources

Resource	Sequences in resource	Sequences found	Percentage found
miRBase (Homo sapiens)	940	453	48.2%
miRBase (Mus musculus)	590	77	13.1%
Homo_sapiens.GRCh37.57.ncrna	12,887	3,586	27.8%

3 Reads

Annotation	Count	Percentage	Perfect matches	%	1 mismatch	%	2 mismatches	%
Annotated	1,511,704	87.9%	1,213,635	80.3%	247,319	16.4%	50,750	3.4%
- with miRBase	1,470,812	97.3%	1,190,140	80.9%	234,618	16.0%	46,054	3.1%
- Homo sapiens	1,436,510	97.7%	1,165,868	81.2%	226,769	15.8%	43,873	3.1%
- Mus musculus	34,302	2.3%	24,272	70.8%	7,849	22.9%	2,181	6.4%
- with Homo_sapiens.GRCh37.57.ncrna	40,892	2.7%	23,495	57.5%	12,701	31.1%	4,696	11.5%
Unannotated	208,537	12.1%						
Total	1,720,241	100.0%						

Figure 19.145: *A summary report of the annotation.*

19.16.4 Working with the small RNA sample

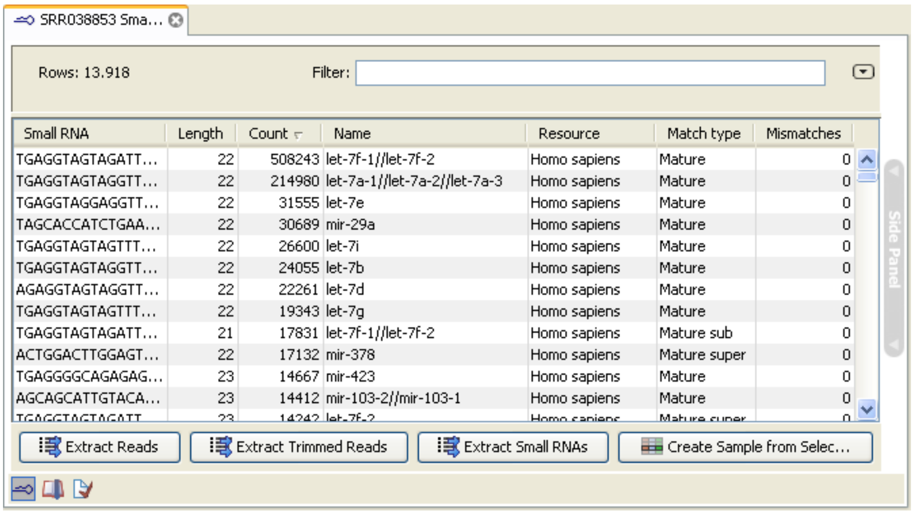
Generally speaking, the small RNA sample comes in two variants:

- The *un-grouped* sample, either as it comes directly from the **Extract and Count** ( ) or when it has been annotated. In this sample, there is one row per tag, and the feature ID is the tag sequence.
- The *grouped* sample created using the **Annotate and Merge Counts** ( ) tool. In this sample, each row represents several tags grouped by a common Mature or Precursor miRNA or other reference.

Below, these two kinds of samples are described in further detail. Note that for both samples, filtering and sorting can be applied, see section C.

The un-grouped sample

An example of an un-grouped annotated sample is shown in figure 19.146.



Small RNA	Length	Count	Name	Resource	Match type	Mismatches
TGAGGTAGTAGATT...	22	508243	let-7f-1//let-7f-2	Homo sapiens	Mature	0
TGAGGTAGTAGGTT...	22	214980	let-7a-1//let-7a-2//let-7a-3	Homo sapiens	Mature	0
TGAGGTAGGAGGTT...	22	31555	let-7e	Homo sapiens	Mature	0
TAGCACCATCTGAA...	22	30689	mir-29a	Homo sapiens	Mature	0
TGAGGTAGTAGTTT...	22	26600	let-7i	Homo sapiens	Mature	0
TGAGGTAGTAGGTT...	22	24055	let-7b	Homo sapiens	Mature	0
AGAGGTAGTAGGTT...	22	22261	let-7d	Homo sapiens	Mature	0
TGAGGTAGTAGTTT...	22	19343	let-7g	Homo sapiens	Mature	0
TGAGGTAGTAGATT...	21	17831	let-7f-1//let-7f-2	Homo sapiens	Mature sub	0
ACTGGACTTGGAGT...	22	17132	mir-378	Homo sapiens	Mature super	0
TGAGGGGCAGAGAG...	23	14667	mir-423	Homo sapiens	Mature	0
AGCAGCATTGTACA...	23	14412	mir-103-2//mir-103-1	Homo sapiens	Mature	0
TGAGGTAGTAGATT	23	14242	let-7f-2	Homo sapiens	Mature super	0

Figure 19.146: *An ungrouped annotated sample.*

By selecting one or more rows in the table, the buttons at the bottom of the view can be used to extract sequences from the table:

Extract Reads (☰) This will extract the original sequencing reads that contributed to this tag. Figure 19.147 shows an example of such a read. The reads include trim annotations (for use when inspecting and double-checking the results of trimming). Note that if these reads are used for read mapping, the trimmed part of the read will automatically be removed. If all rows in the sample are selected and extracted, the sequence list would be the same as the input except for the reads that did not meet the adapter trim settings and the sampling thresholds (tag length and number of copies).

Extract Trimmed Reads (☰) The same as above, except that the trimmed part has been removed.

Extract Small RNAs (☰) This will extract only one copy of each tag.

Note that for all these, you will be able to determine whether a list of DNA or RNA sequences should be produced (when working within the *CLC Genomics Workbench* environment, this only effects the RNA folding tools).

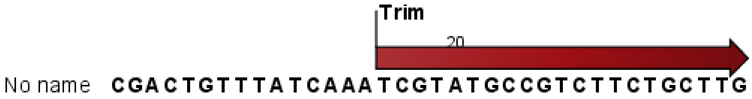
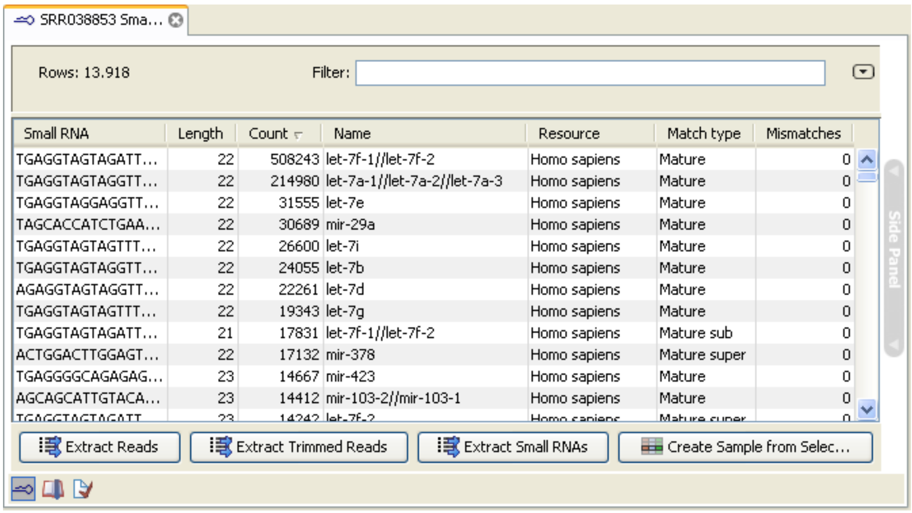


Figure 19.147: *Extracting reads from a sample.*

The button **Create Sample from Selection** (▦) can be used to create a new sample based on the tags that are selected. This can be useful in combination with filtering and sorting.

The grouped sample

An example of a grouped annotated sample is shown in figure 19.148.



Rows: 13.918 Filter:

Small RNA	Length	Count	Name	Resource	Match type	Mismatches
TGAGGTAGTAGATT...	22	508243	let-7f-1//let-7f-2	Homo sapiens	Mature	0
TGAGGTAGTAGGTT...	22	214980	let-7a-1//let-7a-2//let-7a-3	Homo sapiens	Mature	0
TGAGGTAGGAGGTT...	22	31555	let-7e	Homo sapiens	Mature	0
TAGCACCATCTGAA...	22	30689	mir-29a	Homo sapiens	Mature	0
TGAGGTAGTAGTTT...	22	26600	let-7i	Homo sapiens	Mature	0
TGAGGTAGTAGGTT...	22	24055	let-7b	Homo sapiens	Mature	0
AGAGGTAGTAGGTT...	22	22261	let-7d	Homo sapiens	Mature	0
TGAGGTAGTAGTTT...	22	19343	let-7g	Homo sapiens	Mature	0
TGAGGTAGTAGATT...	21	17831	let-7f-1//let-7f-2	Homo sapiens	Mature sub	0
ACTGGACTTGGAGT...	22	17132	mir-378	Homo sapiens	Mature super	0
TGAGGGGCAGAGAG...	23	14667	mir-423	Homo sapiens	Mature	0
AGCAGCATTGTACA...	23	14412	mir-103-2//mir-103-1	Homo sapiens	Mature	0
TGAGGTAGTAGATT	23	14242	let-7f-2	Homo sapiens	Mature super	0

Figure 19.148: *A sample grouped on mature miRNAs.*

The contents of the table are explained in section 19.16.3. In this section, we focus on the tools available for working with the sample.

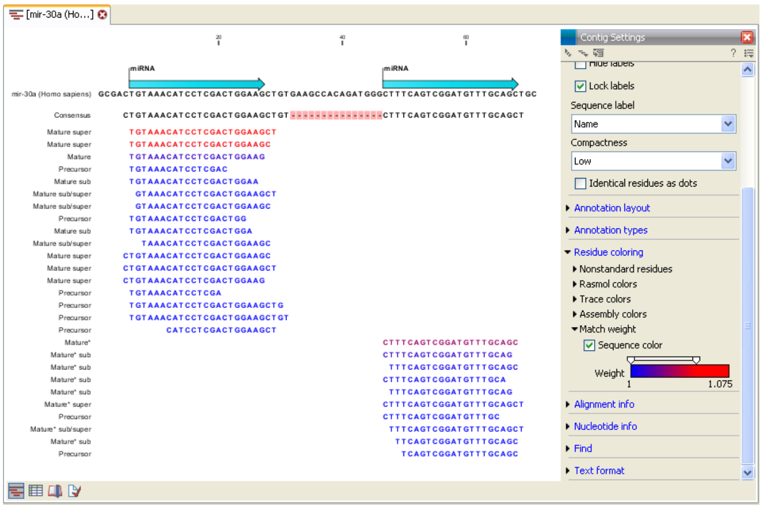
By selecting one or more rows in the table, the buttons at the bottom of the view become active:

Open Read Mapping () This will open a view showing the annotation reference sequence at the top and the tags aligned to it as shown in figure 19.149. The names of the tags indicate their status compared with the reference (e.g. Mature, Mature super, Precursor). This categorization is based on the choices you make when annotating. You can also see the annotations when using miRBase as the annotation source. In this example both the mature and the mature * are annotated, and you can see that both are found in the sample. In the **Side Panel** to the right you can see the **Match weight** group under **Residue coloring** which is used to color the tags according to their relative abundance. The weight is also shown next to the name of the tag. The left side color is used for tags with low counts and the right side color is used for tags with high counts, relative to the total counts of this annotation reference. The sliders just above the gradient color box can be dragged to highlight relevant levels of abundance. The colors can be changed by clicking the box. This will show a list of gradients to choose from.

Create Sample from Selection () This is used to create a new sample based on the tags that are selected. This can be useful in combination with filtering and sorting.

19.16.5 Exploring novel miRNAs

One way of doing this would be to identify interesting tags based on their counts (typically you would be interested in pursuing tags with not too low counts in order to avoid wasting efforts on tags based on reads with sequencing errors), **Extract Small RNAs** () and use this list of tags as input to **Map Reads to Reference** () using the genome as reference. You could then examine where the reads match, and for reads that map in otherwise unannotated regions you could select a region around the match and create a subsequence from this. The subsequence could be folded and examined to see whether the sceondary structure was in agreement with the expected hairpin-type structure for miRNAs.



```
mir-30a (Homo sapiens) GCGACTGTAAACATCCTCGACTGGAAGCTGTGAAGCCACAGATGGGCTTTCAGTCGGATGTTTGCAGCTGC

             Consensus     CTGTAAACATCCTCGACTGGAAGCTGT---------------CTTTCAGTCGGATGTTTGCAGCT
          Mature super      TGTAAACATCCTCGACTGGAAGCT
          Mature super      TGTAAACATCCTCGACTGGAAGC
                Mature      TGTAAACATCCTCGACTGGAAG
             Precursor      TGTAAACATCCTCGAC
            Mature sub      TGTAAACATCCTCGACTGGAA
      Mature sub/super       GTAAACATCCTCGACTGGAAGCT
      Mature sub/super       GTAAACATCCTCGACTGGAAGC
             Precursor      TGTAAACATCCTCGACTGG
            Mature sub      TGTAAACATCCTCGACTGGA
      Mature sub/super        TAAACATCCTCGACTGGAAGC
          Mature super     CTGTAAACATCCTCGACTGGAAGC
          Mature super     CTGTAAACATCCTCGACTGGAAGCT
          Mature super     CTGTAAACATCCTCGACTGGAAG
             Precursor      TGTAAACATCCTCGA
             Precursor      TGTAAACATCCTCGACTGGAAGCTG
             Precursor      TGTAAACATCCTCGACTGGAAGCTGT
             Precursor            CATCCTCGACTGGAAGCT
               Mature*                                                CTTTCAGTCGGATGTTTGCAGC
           Mature* sub                                                CTTTCAGTCGGATGTTTGCAG
           Mature* sub                                                 TTTCAGTCGGATGTTTGCAGC
           Mature* sub                                                CTTTCAGTCGGATGTTTGCA
           Mature* sub                                                 TTTCAGTCGGATGTTTGCAG
         Mature* super                                                CTTTCAGTCGGATGTTTGCAGCT
             Precursor                                                CTTTCAGTCGGATGTTTGC
     Mature* sub/super                                                 TTTCAGTCGGATGTTTGCAGCT
           Mature* sub                                                  TTCAGTCGGATGTTTGCAGC
             Precursor                                                   TCAGTCGGATGTTTGCAGC
```

Contig Settings

□ Hide labels

☑ Lock labels

Sequence label: Name

Compactness: Low

□ Identical residues as dots

Annotation layout

Annotation types

Residue coloring

- Nonstandard residues
- Rasmol colors
- Trace colors
- Assembly colors
- Match weight
  - ☑ Sequence color
  - Weight 1 1.075

Alignment info

Nucleotide info

Find

Text format

Figure 19.149: *Aligning all the variants of this miRNA from miRBase, providing a visual overview of the distribution of tags along the precursor sequence.*

Chapter 20

Expression analysis

Contents

The *CLC Genomics Workbench* is able to analyze expression data produced on microarray platforms and high-throughput sequencing platforms (also known as Next-Generation Sequencing platforms).

Note that the calculation of expression levels based on the raw sequence data is described in section 19.14.

The *CLC Genomics Workbench* provides tools for performing quality control of the data, transformation and normalization, statistical analysis to measure differential expression and annotation-based tests. A number of visualization tools such as volcano plots, MA plots, scatter plots, box plots and heat maps are used to aid the interpretation of the results.

For a hands-on example of how the expression analysis tools work, we suggest taking a look at the tutorial on expression analysis at http://www.clcbio.com/tutorials. Below is a systematic and much more comprehensive description of the various tools.

20.1 Experimental design

In order to make full use of the various tools for interpreting expression data, you need to know the central concepts behind the way the data is organized in the *CLC Genomics Workbench*.

The first piece of data you are faced with is the **sample**. In the Workbench, a sample contains the expression values from either one array or from sequencing data of one sample. Note that the calculation of expression levels based on the raw sequence data is described in sections 19.14 and 19.15 .

See more below on how to get your expression data into the Workbench as samples (under Supported array platforms).

In a sample, there is a number of **features**, usually genes, and their associated expression levels.

To analyze differential expression, you need to tell the workbench how the samples are related. This is done by setting up an **experiment**. An experiment is essentially a set of samples which are grouped. By creating an experiment defining the relationship between the samples, it becomes possible to do statistical analysis to investigate differential expression between the groups. The **Experiment** is also used to accumulate calculations like t-tests and clustering because this information is closely related to the grouping of the samples.

20.1.1 Supported array platforms

The workbench supports analysis of one-color expression arrays. These may be imported from GEO soft sample- or series- file formats, or for Affymetrix arrays, tab-delimited pivot or metrics files, or from Illumina expression files. Expression array data from other platforms may be imported from tab, semi-colon or comma separated files containing the expression feature IDs and levels in a tabular format (see see section K.5).

The workbench assumes that expression values are given at the gene level, thus probe-level analysis of e.g. Affymetrix GeneChips and import of Affymetrix CEL and CDF files is currently not supported. However, the workbench allows import of txt files exported from R containing processed Affymetrix CEL-file data (see see section K.2).

Affymetrix NetAffx annotation files for expression GeneChips in csv format and Illumina annotation files can also be imported. Also, you may import your own annotation data in tabular format see see section K.5).

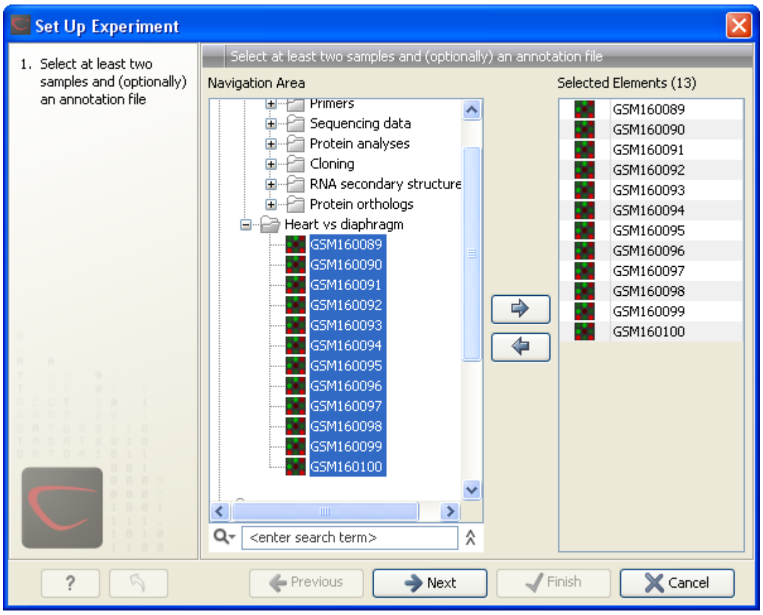
See section K in the Appendix for detailed information about supported file formats.

20.1.2 Setting up an experiment

To set up an experiment:

Toolbox | **Expression Analysis ( )** | **Set Up Experiment ( )**

Select the samples that you wish to use by double-clicking or selecting and pressing the **Add** (➡) button (see figure 20.1).



Set Up Experiment

1. Select at least two samples and (optionally) an annotation file

Select at least two samples and (optionally) an annotation file

Navigation Area

- Primers
- Sequencing data
- Protein analyses
- Cloning
- RNA secondary structure
- Protein orthologs
- Heart vs diaphragm
  - GSM160089
  - GSM160090
  - GSM160091
  - GSM160092
  - GSM160093
  - GSM160094
  - GSM160095
  - GSM160096
  - GSM160097
  - GSM160098
  - GSM160099
  - GSM160100

<enter search term>

Selected Elements (13)

- GSM160089
- GSM160090
- GSM160091
- GSM160092
- GSM160093
- GSM160094
- GSM160095
- GSM160096
- GSM160097
- GSM160098
- GSM160099
- GSM160100

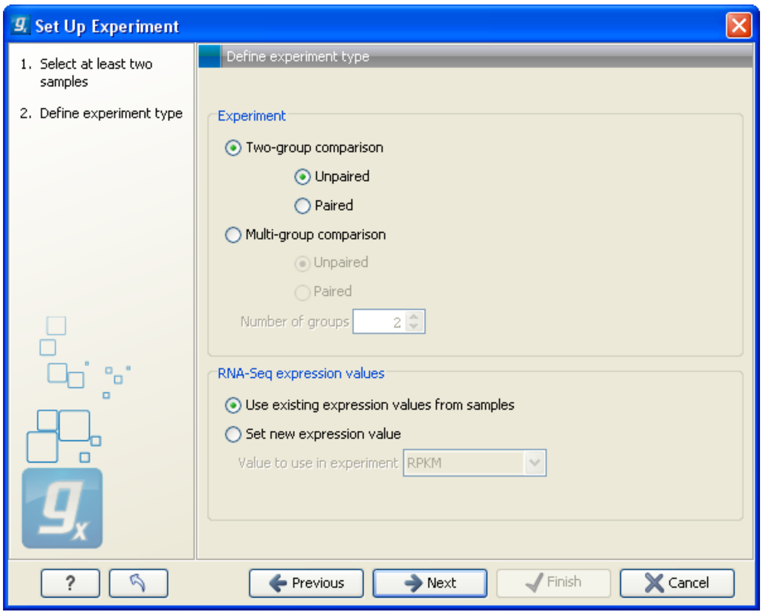
Figure 20.1: *Select the samples to use for setting up the experiment.*

Note that we use "samples" as the general term for both microarray-based sets of expression values and sequencing-based sets of expression values.

Clicking **Next** shows the dialog in figure 20.2.

Here you define the number of groups in the experiment. At the top you can select a two-group experiment, and below you can select a multi-group experiment and define the number of groups.

Note that you can also specify if the samples are paired. Pairing is relevant if you have samples from the same individual under different conditions, e.g. before and after treatment, or at times 0, 2 and 4 hours after treatment. In this case statistical analysis becomes more efficient if effects of the individuals are taken into account, and comparisons are carried out not simply by considering *raw* group means but by considering these *corrected for* effects of the individual. If the **Paired** is selected, a paired rather than a standard t-test will be carried out for two group comparisons. For multiple group comparisons a repeated measures rather than a standard


Set Up Experiment

1. Select at least two samples
2. Define experiment type

Define experiment type

Experiment

● Two-group comparison

● Unpaired

○ Paired

○ Multi-group comparison

● Unpaired

○ Paired

Number of groups 2

RNA-Seq expression values

● Use existing expression values from samples

○ Set new expression value

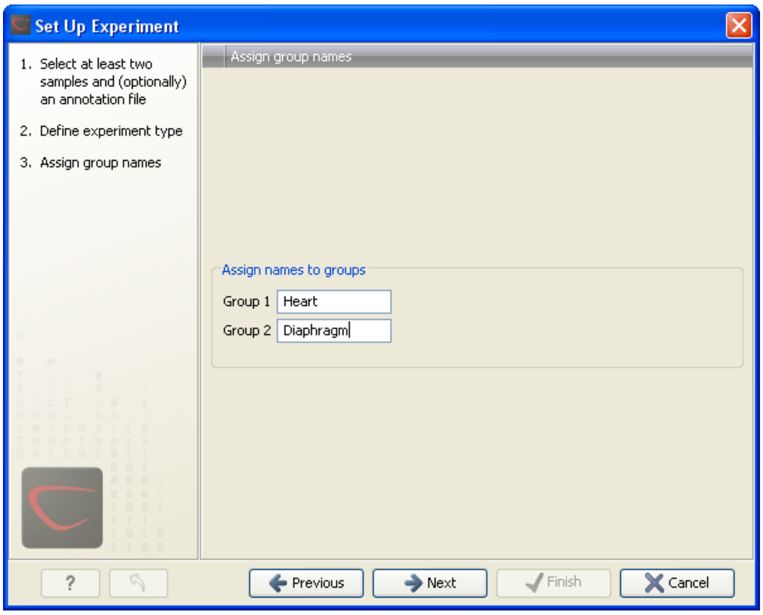
Value to use in experiment RPKM

Figure 20.2: *Defining the number of groups.*

ANOVA will be used.

For RNA-Seq experiments, you can also choose which expression value to be used when setting up the experiment. This value will then be used for all subsequence analyses.

Clicking **Next** shows the dialog in figure 20.3.


Set Up Experiment

1. Select at least two samples and (optionally) an annotation file
2. Define experiment type
3. Assign group names

Assign group names

Assign names to groups

Group 1 Heart

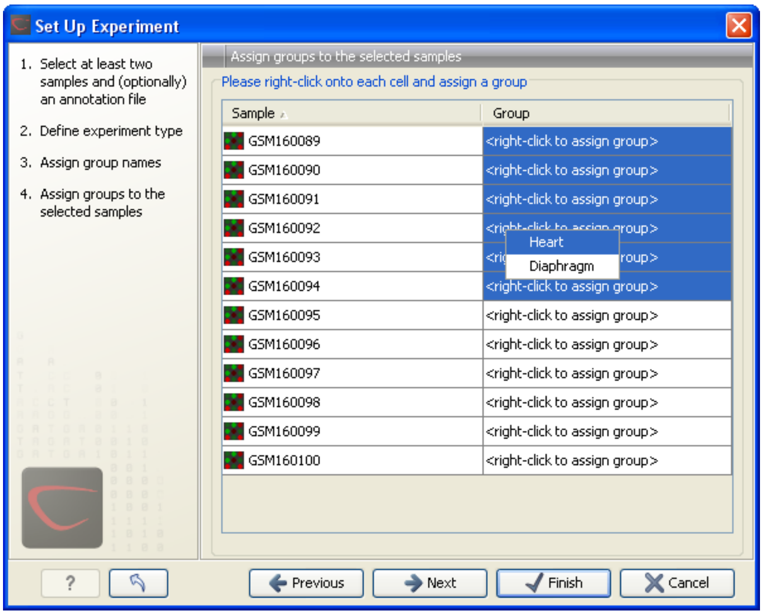
Group 2 Diaphragm

Figure 20.3: *Naming the groups.*

Depending on the number of groups selected in figure 20.2, you will see a list of groups with text fields where you can enter an appropriate name for that group.

For multi-group experiments, if you find out that you have too many groups, click the **Delete** (☒) button. If you need more groups, simply click **Add New Group**.

Click **Next** when you have named the groups, and you will see figure 20.4.



Set Up Experiment

1. Select at least two samples and (optionally) an annotation file
2. Define experiment type
3. Assign group names
4. Assign groups to the selected samples

Assign groups to the selected samples

Please right-click onto each cell and assign a group

Sample	Group
GSM160089	<right-click to assign group>
GSM160090	<right-click to assign group>
GSM160091	<right-click to assign group>
GSM160092	<right-click to assign group>
GSM160093	<ri roup>
GSM160094	<right-click to assign group>
GSM160095	<right-click to assign group>
GSM160096	<right-click to assign group>
GSM160097	<right-click to assign group>
GSM160098	<right-click to assign group>
GSM160099	<right-click to assign group>
GSM160100	<right-click to assign group>

Figure 20.4: *Putting the samples into groups.*

This is where you define which group the individual sample belongs to. Simply select one or more samples (by clicking and dragging the mouse), right-click (Ctrl-click on Mac) and select the appropriate group.

Note that the samples are sorted alphabetically based on their names.

If you have chosen **Paired** in figure 20.2, there will be an extra column where you define which samples belong together. Just as when defining the group membership, you select one or more samples, right-click in the pairing column and select a pair.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

20.1.3 Organization of the experiment table

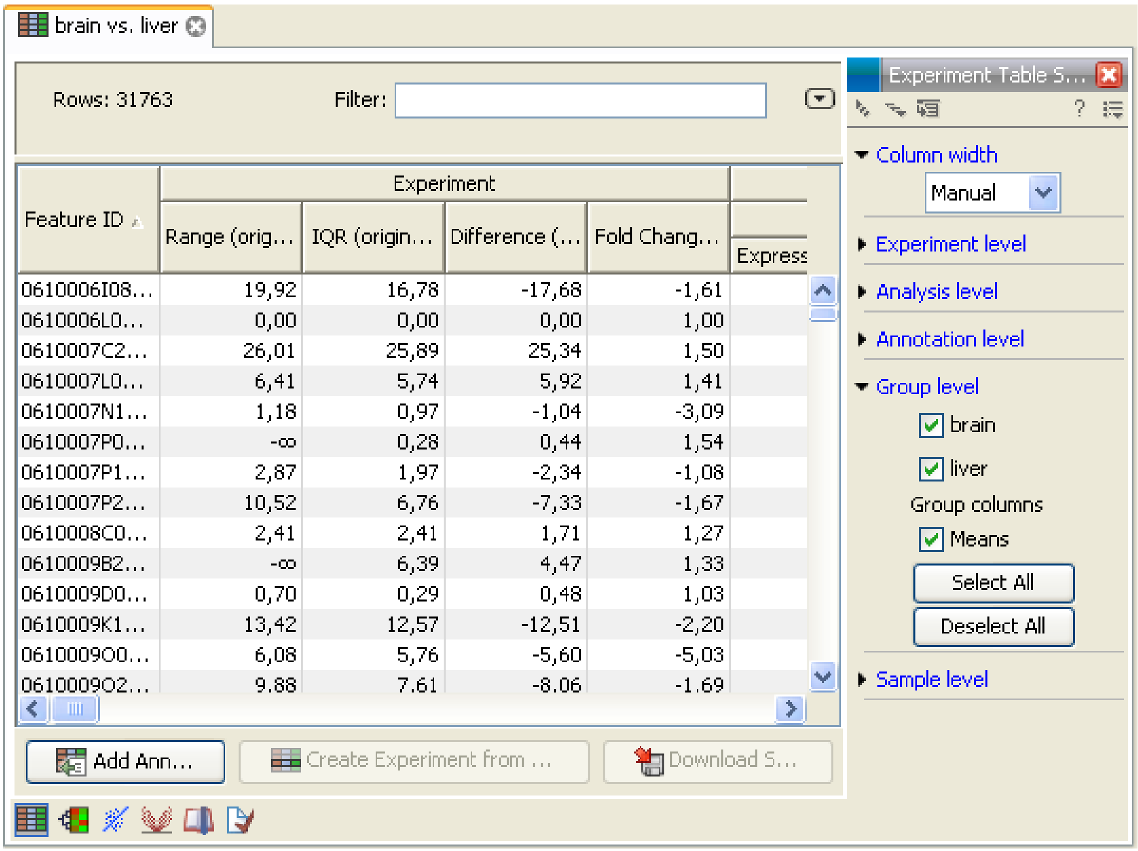
The resulting experiment includes all the expression values and other information from the samples (the values are copied - the original samples are not affected and can thus be deleted with no effect on the experiment). In addition it includes a number of summaries of the values across all, or a subset of, the samples for each feature. Which values are in included is described in the sections below.

When you open it, it is shown in the experiment table (see figure 20.5).

For a general introduction to table features like sorting and filtering, see section C.

Unlike other tables in *CLC Genomics Workbench*, the experiment table has a hierarchical grouping of the columns. This is done to reflect the structure of the data in the experiment. The **Side Panel** is divided into a number of groups corresponding to the structure of the table. These are described below. Note that you can customize and save the settings of the **Side Panel** (see section 5.6).

Whenever you perform analyses like normalization, transformation, statistical analysis etc, new



Rows: 31763 Filter:

Feature ID	Experiment				
	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Express
0610006I08...	19,92	16,78	-17,68	-1,61	
0610006L0...	0,00	0,00	0,00	1,00	
0610007C2...	26,01	25,89	25,34	1,50	
0610007L0...	6,41	5,74	5,92	1,41	
0610007N1...	1,18	0,97	-1,04	-3,09	
0610007P0...	-∞	0,28	0,44	1,54	
0610007P1...	2,87	1,97	-2,34	-1,08	
0610007P2...	10,52	6,76	-7,33	-1,67	
0610008C0...	2,41	2,41	1,71	1,27	
0610009B2...	-∞	6,39	4,47	1,33	
0610009D0...	0,70	0,29	0,48	1,03	
0610009K1...	13,42	12,57	-12,51	-2,20	
0610009O0...	6,08	5,76	-5,60	-5,03	
0610009O2...	9.88	7.61	-8.06	-1.69	

Column width
Manual
Experiment level
Analysis level
Annotation level
Group level
☑ brain
☑ liver
Group columns
☑ Means
Select All
Deselect All
Sample level

Figure 20.5: *Opening the experiment.*

columns will be added to the experiment. You can at any time **Export** ( ) all the data in the experiment in csv or Excel format or **Copy** ( ) the full table or parts of it.

Column width

There are two options to specify the width of the columns and also the entire table:

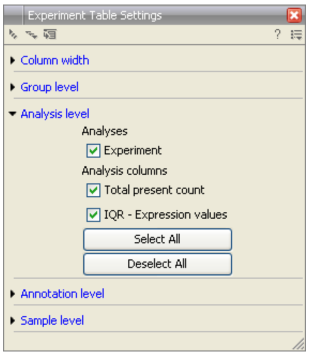
- **Automatic**. This will fit the entire table into the width of the view. This is useful if you only have a few columns.
- **Manual**. This will adjust the width of all columns evenly, and it will make the table as wide as it needs to be to display all the columns. This is useful if you have many columns. In this case there will be a scroll bar at the bottom, and you can manually adjust the width by dragging the column separators.

Experiment level

The rest of the **Side Panel** is devoted to different levels of information on the values in the experiment. The experiment part contains a number of columns that, for each feature ID, provide summaries of the values across all the samples in the experiment (see figure 20.6).

Initially, it has one header for the whole **Experiment**:

- **Total present count**. This is the number of present calls for all samples.



Experiment Table Settings

Column width

Group level

Analysis level

Analyses

☑ Experiment

Analysis columns

☑ Total present count

☑ IQR - Expression values

Select All

Deselect All

Annotation level

Sample level

Figure 20.6: *The initial view of the experiment level for a two-group experiment.*

- **Range (original values)**. The 'Range' column contains the difference between the highest and the lowest expression value for the feature over all the samples. If a feature has the value NaN in one or more of the samples the range value is NaN.

- **IQR (original values)**. The 'IQR' column contains the inter-quantile range of the values for a feature across the samples, that is, the difference between the 75 %-ile value and the 25 %-ile value. For the IQR values, only the numeric values are considered when percentiles are calculated (that is, NaN and +Inf or -Inf values are ignored), and if there are fewer than four samples with numeric values for a feature, the IQR is set to be the difference between the highest and lowest of these.

- **Difference (original values)**. For a two-group experiment the 'Difference' column contains the difference between the mean of the expression values across the samples assigned to group 2 and the mean of the expression values across the samples assigned to group 1. Thus, if the mean expression level in group 2 is higher than that of group 1 the 'Difference' is positive, and if it is lower the 'Difference' is negative. For experiments with more than two groups the 'Difference' contains the difference between the maximum and minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs after the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

- **Fold Change (original values)**. For a two-group experiment the 'Fold Change' tells you how many times bigger the mean expression value in group 2 is relative to that of group 1. If the mean expression value in group 2 is bigger than that in group 1 this value is the mean expression value in group 2 divided by that in group 1. If the mean expression value in group 2 is smaller than that in group 1 the fold change is the mean expression value in group 1 divided by that in group 2 with a negative sign. Thus, if the mean expression levels in group 1 and group 2 are 10 and 50 respectively, the fold change is 5, and if the and if the mean expression levels in group 1 and group 2 are 50 and 10 respectively, the fold change is -5. For experiments with more than two groups, the 'Fold Change' column contains the ratio of the maximum of the mean expression values of the groups to the minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs after the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

Thus, the sign of the values in the 'Difference' and 'Fold change' columns give the direction of the trend across the groups, going from group 1 to group 2, etc.

The columns under the 'Experiment' header are useful for filtering purposes, e.g. you may wish to ignore features that differ too little in expression levels to be confirmed e.g. by qPCR by filtering on the values in the 'Difference', 'IQR' or 'Fold Change' columns or you may wish to ignore features that do not differ at all by filtering on the 'Range' column.

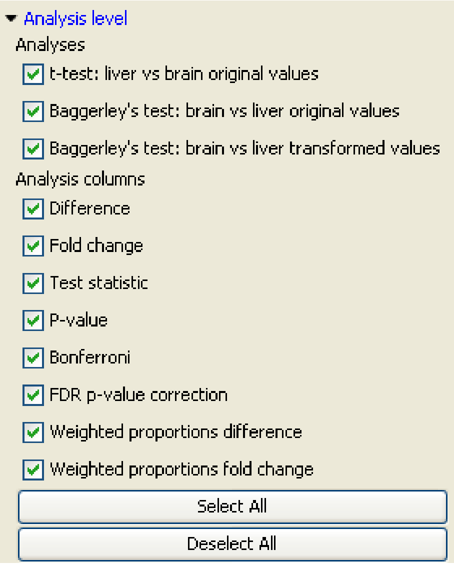
If you have performed normalization or transformation (see sections 20.2.3 and 20.2.2, respectively), the IQR of the normalized and transformed values will also appear. Also, if you later choose to transform or normalize your experiment, columns will be added for the transformed or normalized values.

Note! It is very common to filter features on fold change values in expression analysis and fold change values are also used in volcano plots, see section 20.4.4. There are different definitions of 'Fold Change' in the literature. The definition that is used typically depends on the original scale of the data that is analyzed. For data whose original scale is *not* the log scale the standard definition is the ratio of the group means [Tusher et al., 2001]. This is the value you find in the 'Fold Change' column of the experiment. However, for data whose original *is* the log scale, the difference of the mean expression levels is sometimes referred to as the fold change [Guo et al., 2006], and if you want to filter on fold change for these data you should filter on the values in the 'Difference' column. Your data's original scale will e.g. be the log scale if you have imported Affymetrix expression values which have been created by running the RMA algorithm on the probe-intensities.

Analysis level

If you perform statistical analysis (see section 20.4), there will be a heading for each statistical analysis performed. Under each of these headings you find columns holding relevant values for the analysis (P-value, corrected P-value, test-statistic etc. - see more in section 20.4).

An example of a more elaborate analysis level is shown in figure 20.7.


▾ Analysis level

Analyses

☑ t-test: liver vs brain original values

☑ Baggerley's test: brain vs liver original values

☑ Baggerley's test: brain vs liver transformed values

Analysis columns

☑ Difference

☑ Fold change

☑ Test statistic

☑ P-value

☑ Bonferroni

☑ FDR p-value correction

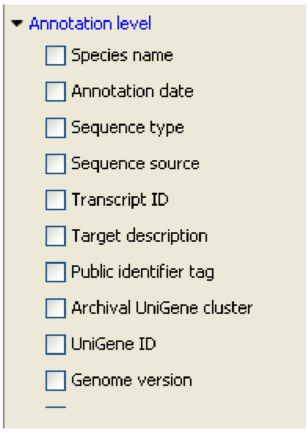
☑ Weighted proportions difference

☑ Weighted proportions fold change

Figure 20.7: *Transformation, normalization and statistical analysis has been performed.*

Annotation level

If your experiment is annotated (see section 20.1.4), the annotations will be listed in the **Annotation level** group as shown in figure 20.8.



▾ Annotation level

□ Species name

□ Annotation date

□ Sequence type

□ Sequence source

□ Transcript ID

□ Target description

□ Public identifier tag

□ Archival UniGene cluster

□ UniGene ID

□ Genome version

Figure 20.8: *An annotated experiment.*

In order to avoid too much detail and cluttering the table, only a few of the columns are shown per default.

Note that if you wish a different set of annotations to be displayed each time you open an experiment, you need to save the settings of the **Side Panel** (see section 5.6).

Group level

At the group level, you can show/hide entire groups (*Heart* and *Diaphragm* in figure 20.5). This will show/hide everything under the group's header. Furthermore, you can show/hide group-level information like the group means and present count within a group. If you have performed normalization or transformation (see sections 20.2.3 and 20.2.2, respectively), the means of the normalized and transformed values will also appear.

Sample level

In this part of the side panel, you can control which columns to be displayed for each sample. Initially this is the **Expression values** and the **Present call**.

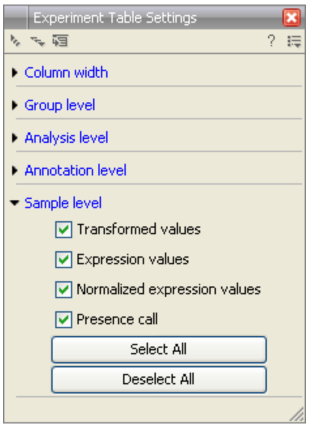
If you have performed normalization or transformation (see sections 20.2.3 and 20.2.2, respectively), the normalized and transformed values will also appear.

An example is shown in figure 20.9.

Creating a sub-experiment from a selection

If you have identified a list of genes that you believe are differentially expressed, you can create a subset of the experiment. (Note that the filtering and sorting may come in handy in this situation, see section C).

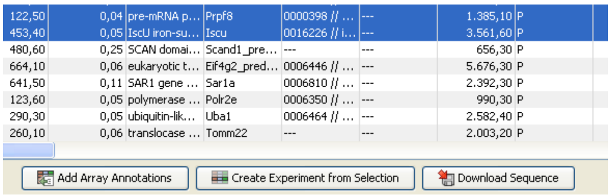
To create a sub-experiment, first select the relevant features (rows). If you have applied a filter and wish to select all the visible features, press Ctrl + A (⌘ + A on Mac). Next, press the **Create**



- ☑ Transformed values
- ☑ Expression values
- ☑ Normalized expression values
- ☑ Presence call

Figure 20.9: *Sample level when transformation and normalization has been performed.*

Experiment from Selection (▦) button at the bottom of the table (see figure 20.10).



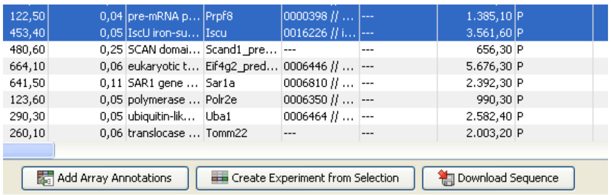
122,50	0,04	pre-mRNA p...	Prpf8	0000398 // ...	---	1.385,10 P
453,40	0,05	IscU iron-su...	Iscu	0016226 // i...	---	3.561,60 P
480,60	0,25	SCAN domai...	Scand1_pre...	---	---	656,30 P
664,10	0,06	eukaryotic t...	Eif4g2_pred...	0006446 // ...	---	5.676,30 P
641,50	0,11	SAR1 gene ...	Sar1a	0006810 // ...	---	2.392,30 P
123,60	0,05	polymerase ...	Polr2e	0006350 // ...	---	990,30 P
290,30	0,05	ubiquitin-lik...	Uba1	0006464 // ...	---	2.582,40 P
260,10	0,06	translocase ...	Tomm22	---	---	2.003,20 P

Figure 20.10: *Create a subset of the experiment by clicking the button at the bottom of the experiment table.*

This will create a new experiment that has the same information as the existing one but with less features.

Downloading sequences from the experiment table

If your experiment is annotated, you will be able to download the GenBank sequence for features which have a GenBank accession number in the 'Public identifier tag' annotation column. To do this, select a number of features (rows) in the experiment and then click **Download Sequence** (💾) (see figure 20.11).



122,50	0,04	pre-mRNA p...	Prpf8	0000398 // ...	---	1.385,10 P
453,40	0,05	IscU iron-su...	Iscu	0016226 // i...	---	3.561,60 P
480,60	0,25	SCAN domai...	Scand1_pre...	---	---	656,30 P
664,10	0,06	eukaryotic t...	Eif4g2_pred...	0006446 // ...	---	5.676,30 P
641,50	0,11	SAR1 gene ...	Sar1a	0006810 // ...	---	2.392,30 P
123,60	0,05	polymerase ...	Polr2e	0006350 // ...	---	990,30 P
290,30	0,05	ubiquitin-lik...	Uba1	0006464 // ...	---	2.582,40 P
260,10	0,06	translocase ...	Tomm22	---	---	2.003,20 P

Figure 20.11: *Select sequences and press the download button.*

This will open a dialog where you specify where the sequences should be saved. You can learn more about opening and viewing sequences in chapter 10. You can now use the downloaded sequences for further analysis in the Workbench, e.g. performing BLAST searches and designing primers for QPCR experiments.

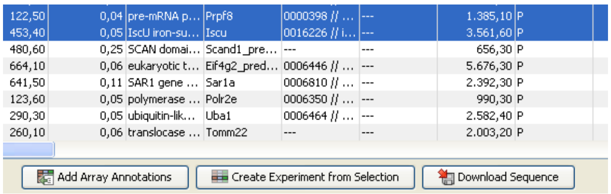
20.1.4 Adding annotations to an experiment

Annotation files provide additional information about each feature. This information could be which GO categories the protein belongs to, which pathways, various transcript and protein identifiers etc. See section K for information about the different annotation file formats that are supported *CLC Genomics Workbench*.

The annotation file can be imported into the Workbench and will get a special icon ( ). See an overview of annotation formats supported by *CLC Genomics Workbench*in section K. In order to associate an annotation file with an experiment, either select the annotation file when you set up the experiment (see section 20.1.2), or click:

Toolbox | Expression Analysis ( ) | Annotation Test | Add Annotations ( )

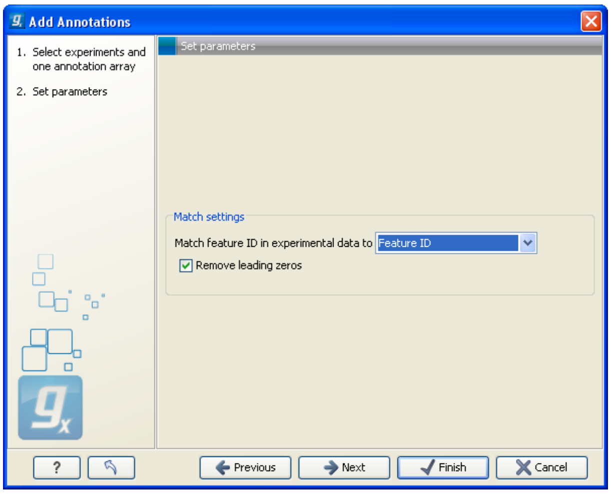
Select the experiment ( ) and the annotation file ( ) and click **Finish**. You will now be able to see the annotations in the experiment as described in section 20.1.3. You can also add annotations by pressing the **Add Annotations** ( ) button at the bottom of the table (see figure 20.12).



122,50	0,04	pre-mRNA p...	Prpf8	0000398 // ...	---	1.385,10	P
453,40	0,05	IscU iron-su...	Iscu	0016226 // i...	---	3.561,60	P
480,60	0,25	SCAN domai...	Scand1_pre...	---	---	656,30	P
664,10	0,06	eukaryotic t...	Eif4g2_pred...	0006446 // ...	---	5.676,30	P
641,50	0,11	SAR1 gene ...	Sar1a	0006810 // ...	---	2.392,30	P
123,60	0,05	polymerase ...	Polr2e	0006350 // ...	---	990,30	P
290,30	0,05	ubiquitin-lik...	Uba1	0006464 // ...	---	2.582,40	P
260,10	0,06	translocase ...	Tomm22	---	---	2.003,20	P

Figure 20.12: *Adding annotations by clicking the button at the bottom of the experiment table.*

This will bring up a dialog where you can select the annotation file that you have imported together with the experiment you wish to annotate. Click **Next** to specify settings as shown in figure 20.13).



Add Annotations

1. Select experiments and one annotation array
2. Set parameters

Set parameters

Match settings

Match feature ID in experimental data to: Feature ID

☑ Remove leading zeros

Figure 20.13: *Choosing how to match annotations with samples.*

In this dialog, you can specify how to match the annotations to the features in the sample. The Workbench looks at the columns in the annotation file and lets you choose which column that should be used for matching to the feature IDs in the experimental data (samples or experiment). Usually the default is right, but for some annotation files, you need to use another column.

Some annotation files have leading zeros in the identifier which you can remove by checking the **Remove leading zeros** box.

Note! Existing annotations on the experiment will be overwritten.

20.1.5 Scatter plot view of an experiment

At the bottom of the experiment table, you can switch between different views of the experiment (see figure 20.14).

Figure 20.14: *An experiment can be viewed in several ways.*

One of the views is the **Scatter Plot** (). The scatter plot can be adjusted to show e.g. the group means for two groups (see more about how to adjust this below).

An example of a scatter plot is shown in figure 20.15.

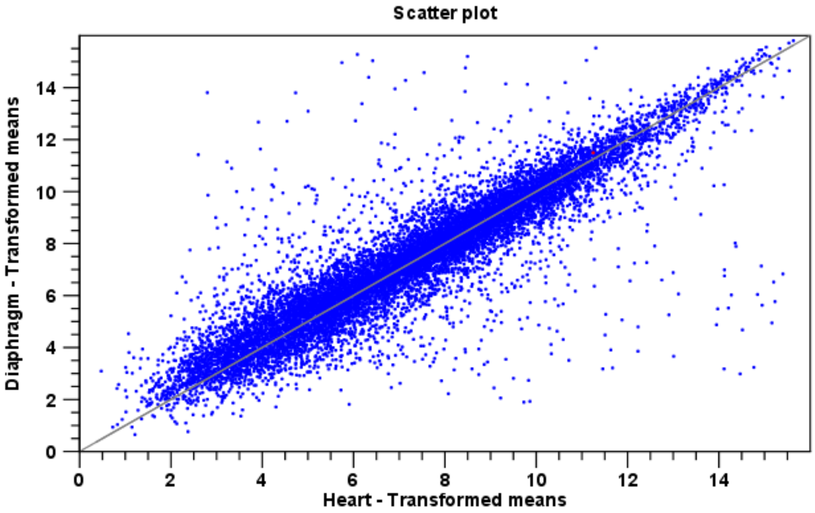


Figure 20.15: *A scatter plot of group means for two groups (transformed expression values).*

In the **Side Panel** to the left, there are a number of options to adjust this view. Under **Graph preferences**, you can adjust the general properties of the scatter plot:

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks

- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Draw x = y axis**. This will draw a diagonal line across the plot. This line is shown per default.
- **Line width**
  - Thin
  - Medium
  - Wide
- **Line type**
  - None
  - Line
  - Long dash
  - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties** preferences, where you can adjust coloring and appearance of the dots:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Finally, the group at the bottom - **Columns to compare** - is where you choose the values to be plotted. Per default for a two-group experiment, the group means are used.

Note that if you wish to use the same settings next time you open a scatter plot, you need to save the settings of the **Side Panel** (see section 5.6).

20.1.6 Cross-view selections

There are a number of different ways of looking at an experiment as shown in figure 20.16).

Figure 20.16: *An experiment can be viewed in several ways.*

Beside the **Experiment table** (▦) which is the default view, the views are: **Scatter plot** (⁘), **Volcano plot** (∨) and the **Heat map** (▦). By pressing and holding the Ctrl (⌘ on Mac) button while you click one of the view buttons in figure 20.16, you can make a split view. This will make it possible to see e.g. the experiment table in one view and the volcano plot in another view.

An example of such a split view is shown in figure 20.17.

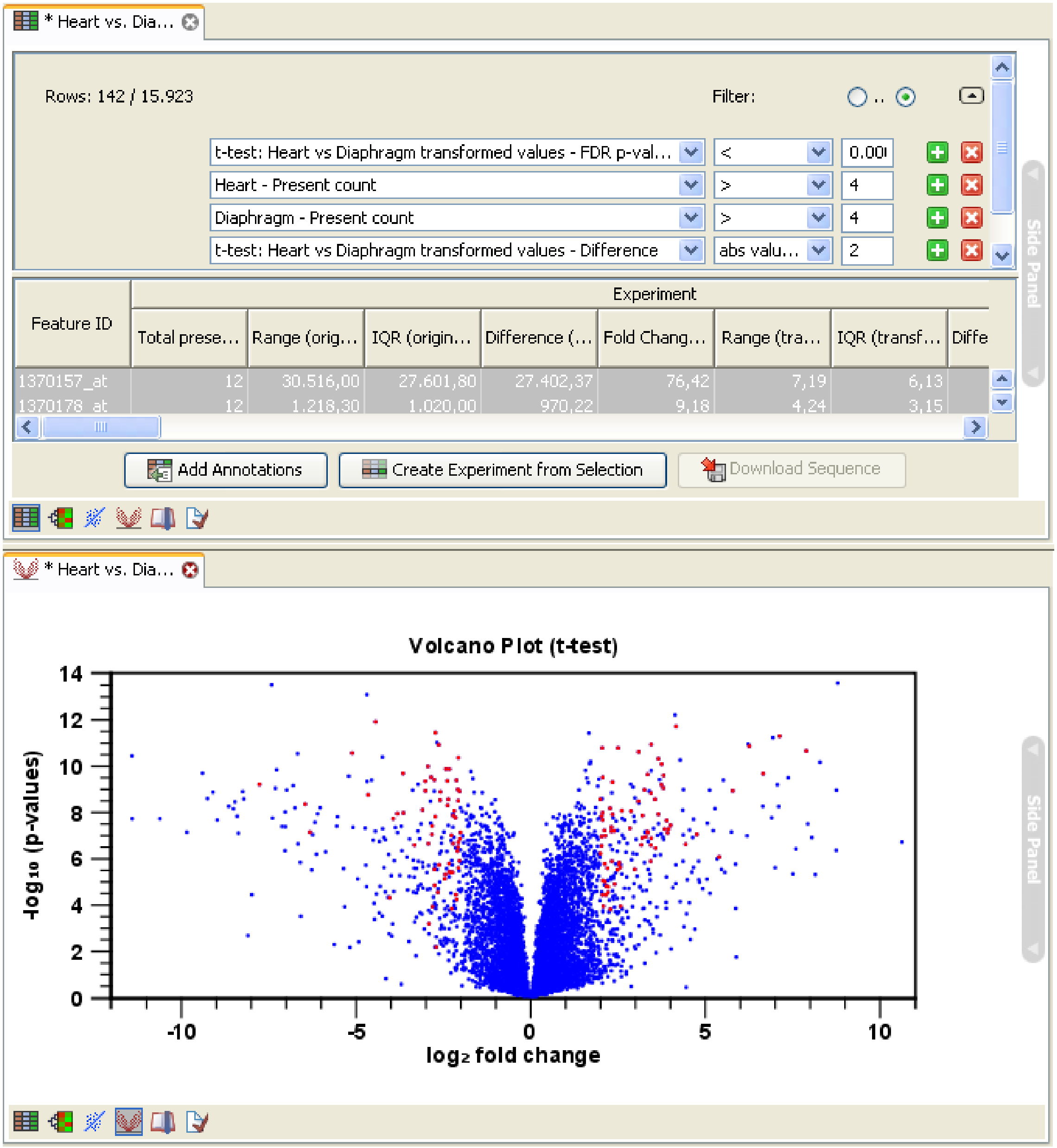
Selections are shared between all these different views of an experiment. This means that if you select a number of rows in the table, the corresponding dots in the scatter plot, volcano plot or heatmap will also be selected. The selection can be made in any view, also the heat map, and all other open views will reflect the selection.

A common use of the split views is where you have an experiment and have performed a statistical analysis. You filter the experiment to identify all genes that have an FDR corrected p-value below 0.05 and a fold change for the test above say, 2. You can select all the rows in the experiment table satisfying these filters by holding down the Cntrl button and clicking 'a'. If you have a split view of the experiment and the volcano plot all points in the volcano plot corresponding to the selected features will be red. Note that the volcano plot allows two sets of values in the columns under the test you are considering to be displayed on the x-axis: the 'Fold change's and the 'Difference's. You control which to plot in the side panel. If you have filtered on 'Fold change' you will typically want to choose 'Fold change' in the side panel. If you have filtered on 'Difference' (e.g. because your original data is on the log scale, see the note on fold change in 20.1.3) you typically want to choose 'Difference'.

20.2 Transformation and normalization

The original expression values often need to be transformed and/or normalized in order to ensure that samples are comparable and assumptions on the data for analysis are met [Allison et al., 2006]. These are essential requirements for carrying out a meaningful analysis. The raw expression values often exhibit a strong dependency of the variance on the mean, and it may be preferable to remove this by log-transforming the data. Furthermore, the sets of expression values in the different samples in an experiment may exhibit systematic differences that are likely due to differences in sample preparation and array processing, rather being the result of the underlying biology. These noise effects should be removed before statistical analysis is carried out.

When you perform transformation and normalization, the original expression values will be kept, and the new values will be added. If you select an experiment (▦), the new values will be added to the experiment (not the original samples). And likewise if you select a sample ((▦) or (☰)) - in this case the new values will be added to the sample (the original values are still kept on the sample).



* Heart vs. Dia...

Rows: 142 / 15.923

Filter: ○ .. ●

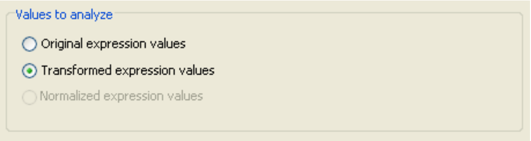
t-test: Heart vs Diaphragm transformed values - FDR p-val...	<	0.001
Heart - Present count	>	4
Diaphragm - Present count	>	4
t-test: Heart vs Diaphragm transformed values - Difference	abs valu...	2

Feature ID	Experiment							
	Total prese...	Range (orig...	IQR (origin...	Difference (...	Fold Chang...	Range (tra...	IQR (transf...	Diffe
1370157_at	12	30.516,00	27.601,80	27.402,37	76,42	7,19	6,13	
1370178_at	12	1.218,30	1.020,00	970,22	9,18	4,24	3,15	

Figure 20.17: *A split view showing an experiment table at the top and a volcano plot at the bottom (note that you need to perform statistical analysis to show a volcano plot, see section 20.4).*

20.2.1 Selecting transformed and normalized values for analysis

A number of the tools in the **Expression Analysis** ( ) folder use expression levels. All of these tools let you choose between *Original*, *Transformed* and *Normalized* expression values as shown in figure 20.18.



Values to analyze

○ Original expression values
● Transformed expression values
○ Normalized expression values

Figure 20.18: *Selecting which version of the expression values to analyze. In this case, the values have not been normalized, so it is not possible to select normalized values.*

In this case, the values have not been normalized, so it is not possible to select normalized values.

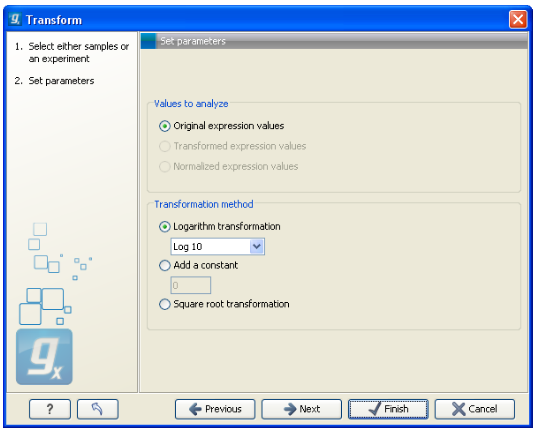
20.2.2 Transformation

The *CLC Genomics Workbench* lets you transform expression values based on logarithm and adding a constant:

> **Toolbox** | **Expression Analysis ( )** | **Transformation and Normalization** | **Transform ( )**

Select a number of samples (( ) or ( )) or an experiment ( ) and click **Next**.

This will display a dialog as shown in figure 20.19.



Transform

1. Select either samples or an experiment
2. Set parameters

Set parameters

Values to analyze

● Original expression values
○ Transformed expression values
○ Normalized expression values

Transformation method

● Logarithm transformation
Log 10
○ Add a constant
0
○ Square root transformation

Figure 20.19: *Transforming expression values.*

At the top, you can select which values to transform (see section 20.2.1).

Next, you can choose three kinds of transformation:

- **Logarithm transformation**. Transformed expression values will be calculated by taking the logarithm (of the specified type) of the values you have chosen to transform.
  - **10**.
  - **2**.
  - **Natural logarithm**.
- **Adding a constant**. Transformed expression values will be calculated by adding the specified constant to the values you have chosen to transform.
- **Square root transformation**.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

20.2.3 Normalization

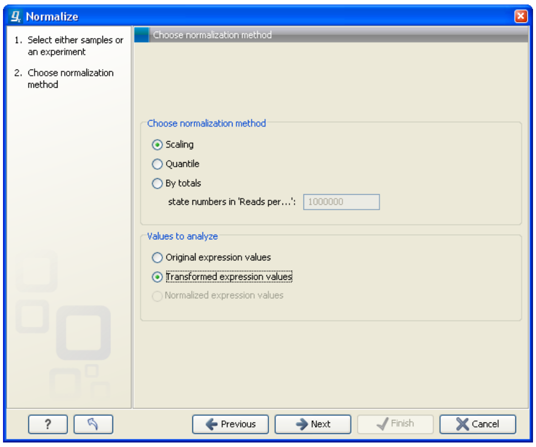
The *CLC Genomics Workbench* lets you normalize expression values.

To start the normalization:

Toolbox | **Expression Analysis** (**)** | **Transformation and Normalization** | **Normalize** (**)**

Select a number of samples (() or ()) or an experiment () and click **Next**.

This will display a dialog as shown in figure 20.20.



Normalize

1. Select either samples or an experiment
2. Choose normalization method

Choose normalization method

Choose normalization method

● Scaling

○ Quantile

○ By totals

state numbers in 'Reads per...': 1000000

Values to analyze

○ Original expression values

● Transformed expression values

○ Normalized expression values

Figure 20.20: *Choosing normalization method.*

At the top, you can choose three kinds of normalization (for mathematical descriptions see [Bolstad et al., 2003]):

- **Scaling**. The sets of the expression values for the samples will be multiplied by a constant so that the sets of normalized values for the samples have the same 'target' value (see description of the **Normalization value** below).
- **Quantile**. The empirical distributions of the sets of expression values for the samples are used to calculate a common target distribution, which is used to calculate normalized sets of expression values for the samples.
- **By totals**. This option is intended to be used with count-based data, i.e. data from RNA-seq, small RNA or expression profiling by tags. A sum is calculated for the expression values in a sample. The transformed value are generated by dividing the input values by the sample sum and multiplying by the factor (e.g. per '1,000,000').

Figures 20.21 and 20.22 show the effect on the distribution of expression values when using scaling or quantile normalization, respectively.

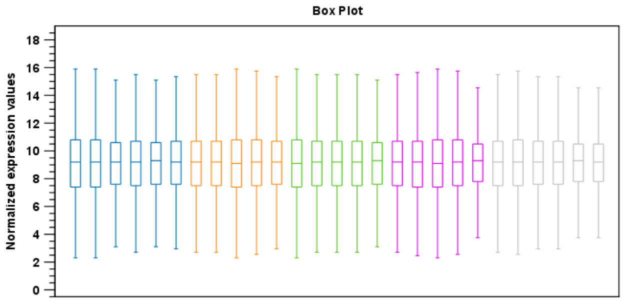


Figure 20.21: *Box plot after scaling normalization.*

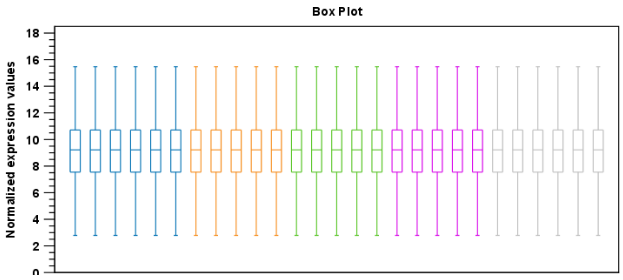
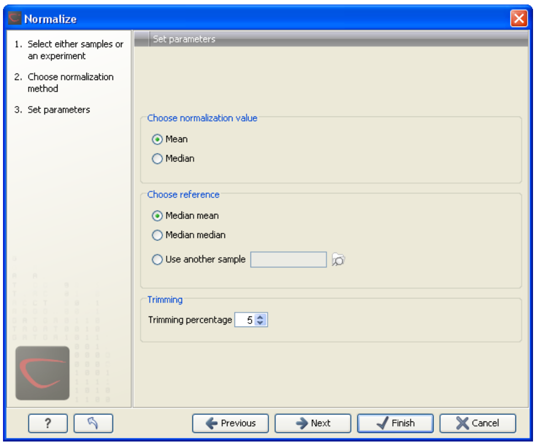


Figure 20.22: *Box plot after quantile normalization.*

At the bottom of the dialog in figure 20.20, you can select which values to normalize (see section 20.2.1).

Clicking **Next** will display a dialog as shown in figure 20.23.


Normalize
1. Select either samples or an experiment
2. Choose normalization method
3. Set parameters
Set parameters
?
Previous
Next
Finish
Cancel

Choose normalization value

● Mean

○ Median

Choose reference

● Median mean

○ Median median

○ Use another sample

Trimming

Trimming percentage 5

Figure 20.23: *Normalization settings.*

The following parameters can be set:

- **Normalization value.** The type of value of the samples which you want to ensure are equal for the normalized expression values
  - **Mean.**
  - **Median.**
- **Reference.** The specific value that you want the normalized value to be after normalization.
  - **Median mean.**
  - **Median median.**
  - **Use another sample.**
- **Trimming percentage.** Expression values that lie below the value of this percentile, or above 100 minus the value of this percentile, in the empirical distribution of the expression values in a sample will be excluded when calculating the normalization and reference values.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

20.3 Quality control

The *CLC Genomics Workbench* includes a number of tools for quality control. These allow visual inspection of the overall distributions, variability and similarity of the sets of expression values in samples, and may be used to spot unwanted systematic differences between samples, outlying samples and samples of poor quality, that you may want to exclude.

20.3.1 Creating box plots - analyzing distributions

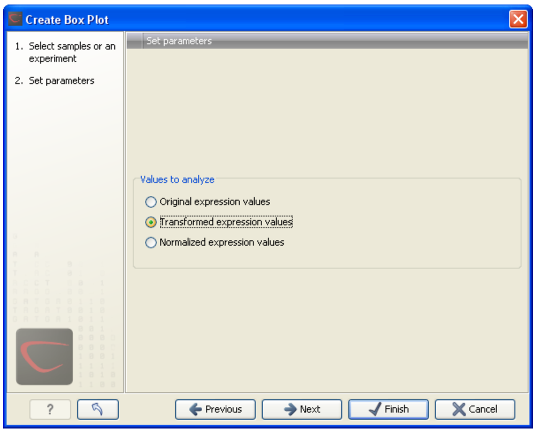
In most cases you expect the majority of genes to behave similarly under the conditions considered, and only a smaller proportion to behave differently. Thus, at an overall level you would expect the distributions of the sets of expression values in samples in a study to be similar. A boxplot provides a visual presentation of the distributions of expression values in samples. For each sample the distribution of it's values is presented by a line representing a center, a box representing the middle part, and whiskers representing the tails of the distribution. Differences in the overall distributions of the samples in a study may indicate that normalization is required before the samples are comparable. An atypical distribution for a single sample (or a few samples), relative to the remaining samples in a study, could be due to imperfections in the preparation and processing of the sample, and may lead you to reconsider using the sample(s).

To create a box plot:

> **Toolbox** | **Expression Analysis ( )** | **Quality Control** | **Create Box Plot ( )**

Select a number of samples (( ) or ( )) or an experiment ( ) and click **Next**.

This will display a dialog as shown in figure 20.24.



Create Box Plot

1. Select samples or an experiment
2. Set parameters

Set parameters

Values to analyze

○ Original expression values
● Transformed expression values
○ Normalized expression values

Figure 20.24: *Choosing values to analyze for the box plot.*

Here you select which values to use in the box plot (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Viewing box plots

An example of a box plot of a two-group experiment with 12 samples is shown in figure 20.25.

Note that the boxes per default are colored according to their group relationship. At the bottom you find the names of the samples, and the y-axis shows the expression values (note that sample

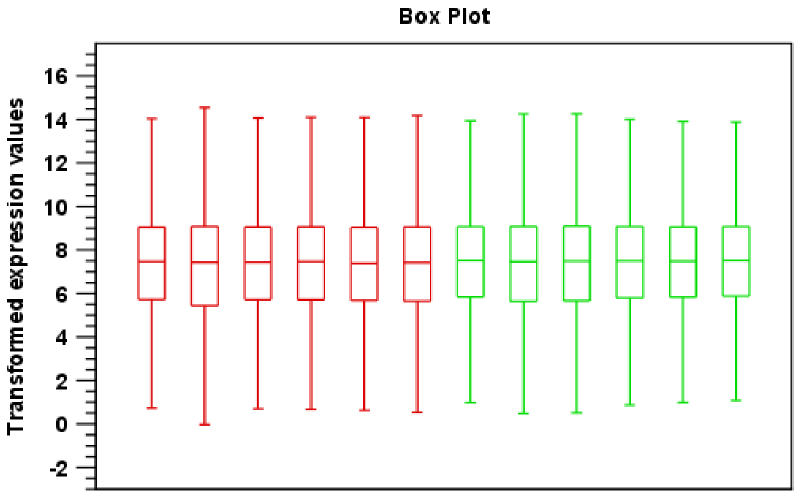
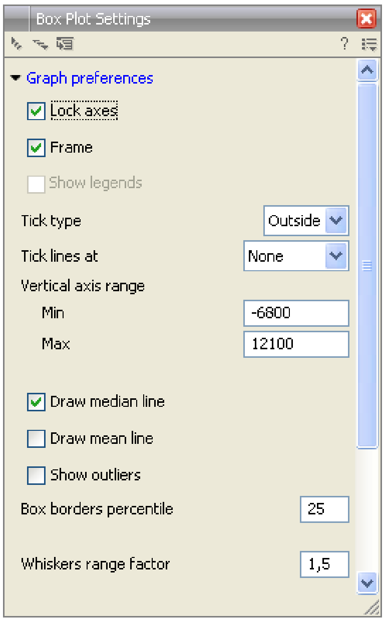


Figure 20.25: *A box plot of 12 samples in a two-group experiment, colored by group.*

names are not shown in figure 20.25).

Per default the box includes the IQR values (from the lower to the upper quartile), the median is displayed as a line in the box, and the whiskers extend 1.5 times the height of the box.

In the **Side Panel** to the left, there is a number of options to adjust this view. Under **Graph preferences**, you can adjust the general properties of the box plot (see figure 20.26).



Box Plot Settings

Graph preferences

☑ Lock axes

☑ Frame

□ Show legends

Tick type: Outside

Tick lines at: None

Vertical axis range

Min: -6800

Max: 12100

☑ Draw median line

□ Draw mean line

□ Show outliers

Box borders percentile: 25

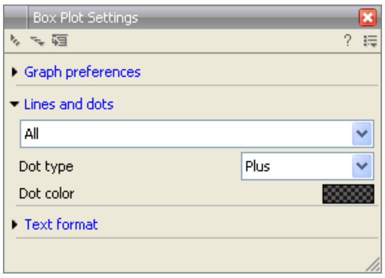
Whiskers range factor: 1,5

Figure 20.26: *Graph preferences for a box plot.*

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.

- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Draw median line**. This is the default - the median is drawn as a line in the box.
- **Draw mean line**. Alternatively, you can also display the mean value as a line.
- **Show outliers**. The values outside the whiskers range are called outliers. Per default they are not shown. Note that the dot type that can be set below only takes effect when outliers are shown. When you select and deselect the **Show outliers**, the vertical axis range is automatically re-calculated to accommodate the new values.

Below the general preferences, you find the **Lines and dots** preferences, where you can adjust coloring and appearance (see figure 20.27).



All
Dot type: Plus
Dot color: ▨

Figure 20.27: *Lines and dot preferences for a box plot.*

- **Select sample or group**. When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.
- **Dot type**
  - None
  - Cross
  - Plus
  - Square

- Diamond
- Circle
- Triangle
- Reverse triangle
- Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a box plot, you need to save the settings of the **Side Panel** (see section 5.6).

Interpreting the box plot

This section will show how to interpret a box plot through a few examples.

First, if you look at figure 20.28, you can see a box plot for an experiment with 5 groups and 27 samples.

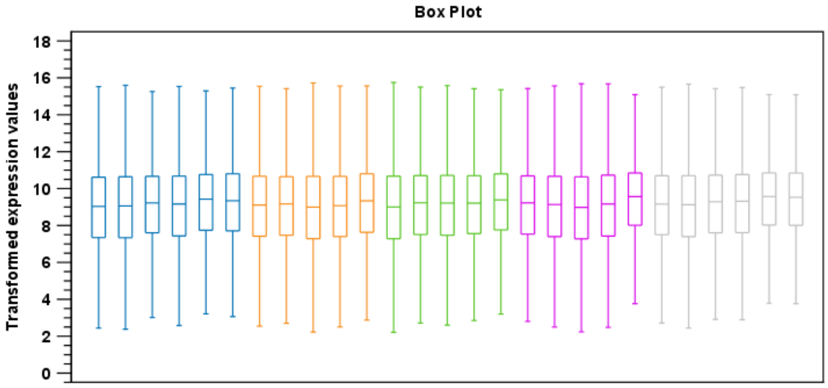


Figure 20.28: *Box plot for an experiment with 5 groups and 27 samples.*

None of the samples stand out as having distributions that are atypical: the boxes and whiskers ranges are about equally sized. The locations of the distributions however, differ some, and indicate that normalization may be required. Figure 20.29 shows a box plot for the same experiment after quantile normalization: the distributions have been brought into par.

In figure 20.30 a box plot for a two group experiment with 5 samples in each group is shown.

The distribution of values in the second sample from the left is quite different from those of other samples, and could indicate that the sample should not be used.

20.3.2 Hierarchical clustering of samples

A hierarchical clustering of samples is a tree representation of their relative similarity. The tree structure is generated by

1. letting each feature be a cluster

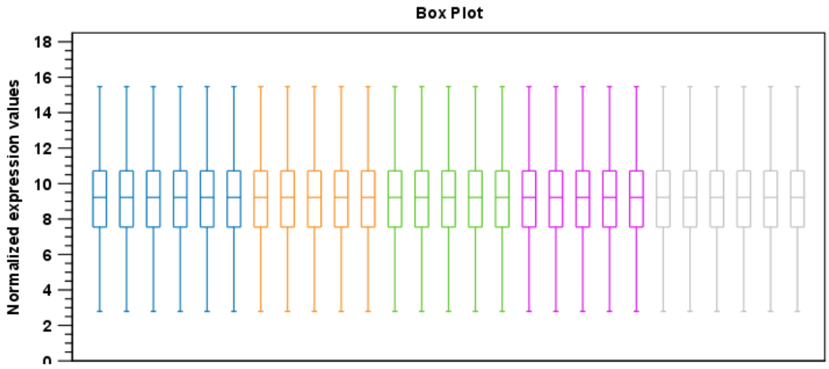


Figure 20.29: *Box plot after quantile normalization.*

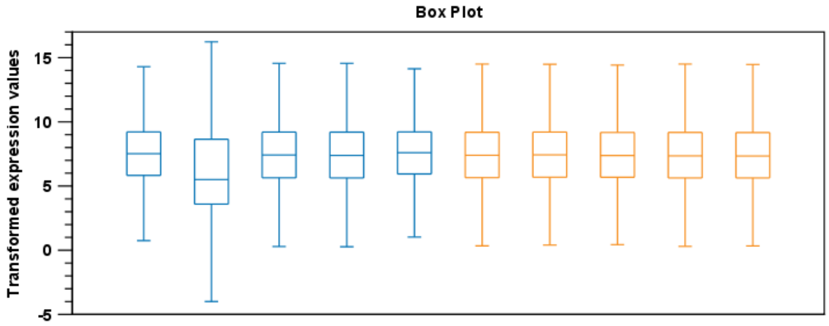


Figure 20.30: *Box plot for a two-group experiment with 5 samples.*

2. calculating pairwise distances between all clusters
3. joining the two closest clusters into one new cluster
4. iterating 2-3 until there is only one cluster left (which will contain all samples).

The tree is drawn so that the distances between clusters are reflected by the lengths of the branches in the tree. Thus, features with expression profiles that closely resemble each other have short distances between them, those that are more different, are placed further apart. (See [Eisen et al., 1998] for a classical example of application of a hierarchical clustering algorithm in microarray analysis. The example is on features rather than samples).

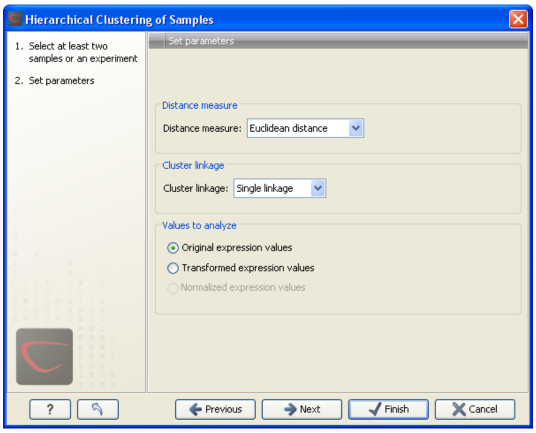
To start the clustering:

> **Toolbox** | **Expression Analysis** ( ) | **Quality Control** | **Hierarchical Clustering of Samples** ( )

Select a number of samples (( ) or ( )) or an experiment ( ) and click **Next**.

This will display a dialog as shown in figure 20.31. The hierarchical clustering algorithm requires that you specify a distance measure and a cluster linkage. The similarity measure is used to specify how distances between two samples should be calculated. The cluster distance metric specifies how you want the distance between two clusters, each consisting of a number of samples, to be calculated.

At the top, you can choose three kinds of **Distance measures**:



Hierarchical Clustering of Samples

1. Select at least two samples or an experiment
2. Set parameters

Set parameters

Distance measure

Distance measure: Euclidean distance

Cluster linkage

Cluster linkage: Single linkage

Values to analyze

● Original expression values
○ Transformed expression values
○ Normalized expression values

Figure 20.31: *Parameters for hierarchical clustering of samples.*

- **Euclidean distance**. The ordinary distance between two points - the length of the segment connecting them. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean distance between u and v is

$$|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.$$

- **1 - Pearson correlation**. The Pearson correlation coefficient between two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ is defined as

$$r = \frac{1}{n-1} \sum_{i=1}^{n} (\frac{x_i - \overline{x}}{s_x}) * (\frac{y_i - \overline{y}}{s_y})$$

  where $\overline{x}/\overline{y}$ is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance**. The Manhattan distance between two points is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between u and v is

$$|u - v| = \sum_{i=1}^{n} |u_i - v_i|.$$

Next, you can select the cluster linkage to be used:

- **Single linkage**. The distance between two clusters is computed as the distance between the two closest elements in the two clusters.

- **Average linkage**. The distance between two clusters is computed as the average distance between objects from the first cluster and objects from the second cluster. The averaging is performed over all pairs (x, y), where x is an object from the first cluster and y is an object from the second cluster.

- **Complete linkage**. The distance between two clusters is computed as the maximal object-to-object distance $d(x_i, y_j)$, where x_i comes from the first cluster, and y_j comes from the second cluster. In other words, the distance between two clusters is computed as the distance between the two farthest objects in the two clusters.

At the bottom, you can select which values to cluster (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Result of hierarchical clustering of samples

The result of a sample clustering is shown in figure 20.32.

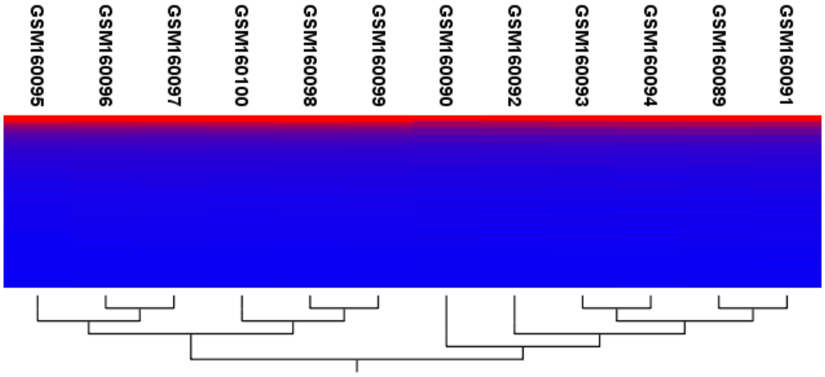


Figure 20.32: *Sample clustering.*

If you have used an **experiment** (▦) as input, the clustering is added to the experiment and will be saved when you save the experiment. It can be viewed by clicking the **Show Heat Map** (▦) button at the bottom of the view (see figure 20.33).

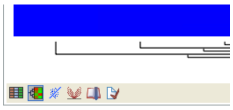


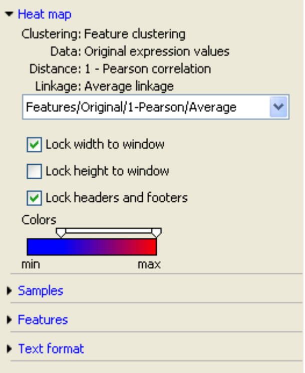
Figure 20.33: *Showing the hierarchical clustering of an experiment.*

If you have selected a number of **samples** ((▦) or (▤)) as input, a new element will be created that has to be saved separately.

Regardless of the input, the view of the clustering is the same. As you can see in figure 20.32, there is a tree at the bottom of the view to visualize the clustering. The names of the samples are listed at the top. The features are represented as horizontal lines, colored according to the expression level. If you place the mouse on one of the lines, you will see the names of the feature to the left. The features are sorted by their expression level in the first sample (in order to cluster the features, see section 20.5.1).

Researchers often have a priori knowledge of which samples in a study should be similar (e.g. samples from the same experimental condition) and which should be different (samples from biological distinct conditions). Thus, researches have expectations about how they should cluster. Samples that are placed unexpectedly in the hierarchical clustering tree may be samples that have been wrongly allocated to a group, samples of unintended or unclean tissue composition or samples for which the processing has gone wrong. Unexpectedly placed samples, of course, could also be highly interesting samples.

There are a number of options to change the appearance of the heat map. At the top of the **Side Panel**, you find the **Heat map** preference group (see figure 20.34).



Heat map

Clustering: Feature clustering

Data: Original expression values

Distance: 1 - Pearson correlation

Linkage: Average linkage

Features/Original/1-Pearson/Average

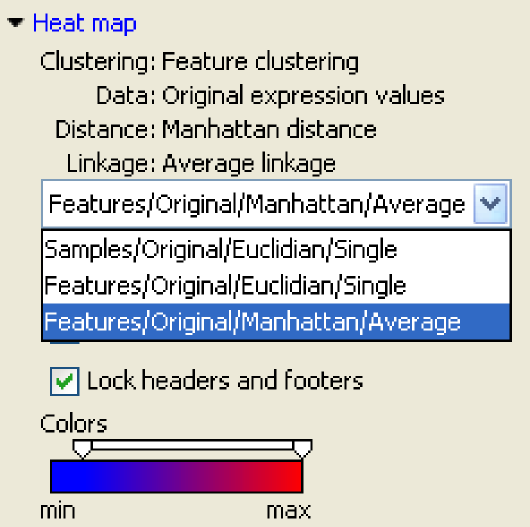
☑ Lock width to window

□ Lock height to window

☑ Lock headers and footers

Figure 20.34: *Side Panel of heat map.*

At the top, there is information about the heat map currently displayed. The information regards type of clustering, expression value used together with distance and linkage information. If you have performed more than one clustering, you can choose between the resulting heat maps in a drop-down box (see figure 20.46).



Heat map

Clustering: Feature clustering

Data: Original expression values

Distance: Manhattan distance

Linkage: Average linkage

Features/Original/Manhattan/Average

Samples/Original/Euclidian/Single
Features/Original/Euclidian/Single
Features/Original/Manhattan/Average

☑ Lock headers and footers

Figure 20.35: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

Note that if you perform an identical clustering, the existing heat map will simply be replaced. Below this box, there is a number of settings for displaying the heat map.

- **Lock width to window**. When you zoom in the heat map, you will per default only zoom in on the vertical level. This is because the width of the heat map is locked to the window. If you uncheck this option, you will zoom both vertically and horizontally. Since you always have more features than samples, it is useful to lock the width since you then have all the samples in view all the time.

- **Lock height to window**. This is the corresponding option for the height. Note that if you check both options, you will not be able to zoom at all, since both the width and the height is fixed.
- **Lock headers and footers**. This will ensure that you are always able to see the sample and feature names and the trees when you zoom in.
- **Colors**. The expression levels are visualized using a gradient color scheme, where the right side color is used for high expression levels and the left side color is used for low expression levels. You can change the coloring by clicking the box, and you can change the relative coloring of the values by dragging the two knobs on the white slider above.

Below you find the **Samples** and **Features** groups. They contain options to show names above/below and left/right, respectively. Furthermore, they contain options to show the tree above/below or left/right, respectively. Note that for clustering of samples, you find the tree options in the **Samples** group, and for clustering of features, you find the tree options in the **Features** group. With the tree options, you can also control the **Tree size**, from tiny to very large, and the option of showing the full tree, no matter how much space it will use.

Note that if you wish to use the same settings next time you open a heat map, you need to save the settings of the **Side Panel** (see section 5.6).

20.3.3 Principal component analysis

A principal component analysis is a mathematical analysis that identifies and quantifies the directions of variability in the data. For a set of samples, e.g. an experiment, this can be done by finding the eigenvectors and eigenvalues of the covariance matrix of the samples. The eigenvectors are orthogonal. The first principal component is the eigenvector with the largest eigenvalue, and specifies the direction with the largest variability. The second principal component is the eigenvector with the second largest eigenvalue, and specifies the direction with the second largest variability. Similarly for the third, etc. The data can be projected onto the space spanned by the eigenvectors. A plot of the data in the space spanned by the first and second principal component will show a simplified version of the data with variability in other directions than the two major directions of variability ignored.

To start the analysis:

Toolbox | Expression Analysis () | Quality Control | Principal Component Analysis ()

Select a number of samples (() or ()) or an experiment () and click **Next**.

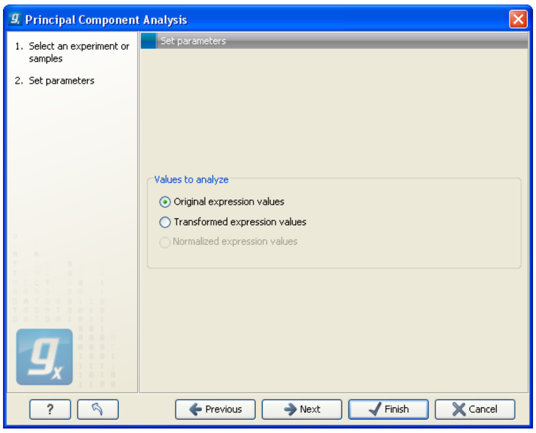
This will display a dialog as shown in figure 20.36.

In this dialog, you select the values to be used for the principal component analysis (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Principal component analysis plot

This will create a principal component plot as shown in figure 20.37.



Principal Component Analysis

1. Select an experiment or samples
2. Set parameters

Set parameters

Values to analyze

- ◉ Original expression values
- ○ Transformed expression values
- ○ Normalized expression values

Figure 20.36: *Selcting which values the principal component analysis should be based on.*

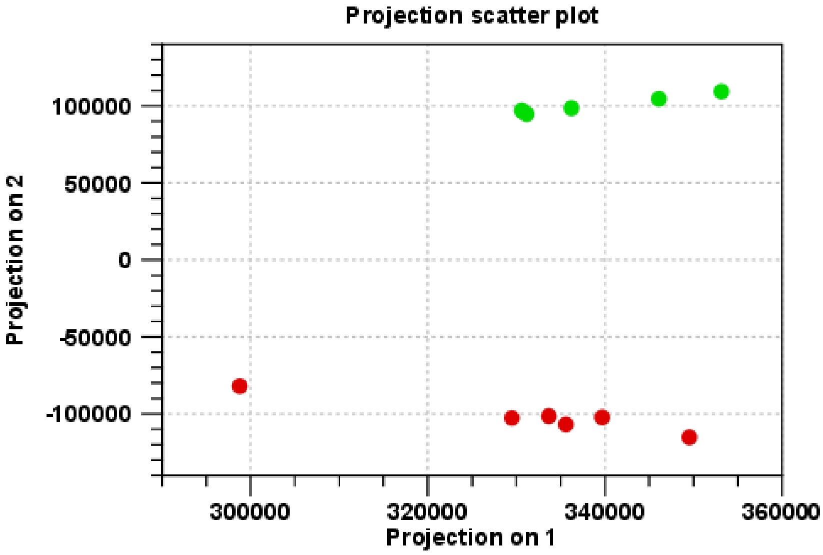


Figure 20.37: *A principal component analysis colored by group.*

The plot shows the projection of the samples onto the two-dimensional space spanned by the first and second principal component. (These are the orthogonal directions in which the data exhibits the largest and second-largest variability).

The plot in figure 20.37 is based on a two-group experiment. The group relationships are indicated by color. We expect the samples from within a group to exhibit less variability when compared, than samples from different groups. Thus samples should cluster according to groups and this is what we see. The PCA plot is thus helpful in identifying outlying samples and samples that have been wrongly assigned to a group.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.

- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **y = 0 axis**. Draws a line where y = 0. Below there are some options to control the appearance of the line:
  - **Line width**
    - Thin
    - Medium
    - Wide
  - **Line type**
    - None
    - Line
    - Long dash
    - Short dash
  - **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties**:

- **Select sample or group**. When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.
- **Dot type**
  - None
  - Cross

  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Show name**. This will show a label with the name of the sample next to the dot. Note that the labels quickly get crowded, so that is why the names are not put on per default.

Note that if you wish to use the same settings next time you open a principal component plot, you need to save the settings of the **Side Panel** (see section 5.6).

Scree plot

Besides the view shown in figure 20.37, the result of the principal component can also be viewed as a scree plot by clicking the **Show Scree Plot** (📊) button at the bottom of the view. The scree plot shows the proportion of variation in the data explained by the each of the principal components. The first principal component explains about 99 percent of the variability.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

The **Lines and plots** below contains the following parameters:

- **Dot type**
    - None
    - Cross
    - Plus
    - Square
    - Diamond
    - Circle
    - Triangle
    - Reverse triangle
    - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Line width**
    - Thin
    - Medium
    - Wide
- **Line type**
    - None
    - Line
    - Long dash
    - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that the graph title and the axes titles can be edited simply by clicking them with the mouse. These changes will be saved when you **Save** ( ) the graph - whereas the changes in the **Side Panel** need to be saved explicitly (see section 5.6).

20.4 Statistical analysis - identifying differential expression

The *CLC Genomics Workbench* is designed to help you identify differential expression. You have a choice of a number of standard statistical tests, that are suitable for different data types and different types of experimental settings. There are two main categories of tests: tests that assume that the data has Gaussian distributions and compare means (described in section 20.4.1) and tests that compare proportions and assume that data consists of counts and (described in section 20.4.2). To run the statistical analysis:

Toolbox | Expression Analysis ( ) | Statistical Analysis | On Gaussian Data ( )

or **Toolbox** | **Expression Analysis** ( ) | **Statistical Analysis** | **On Proportions** ( )

For both kinds of statistics you first select the experiment ( ) that you wish to use and click **Next** (learn more about setting up experiments in section 20.1.2).

The first part of the explanation of how to proceed and perform the statistical analysis is divided into two, depending on whether you are doing Gaussian-based tests or tests on proportions. The last part has an explanation of the options regarding corrected p-values which applies to all tests.

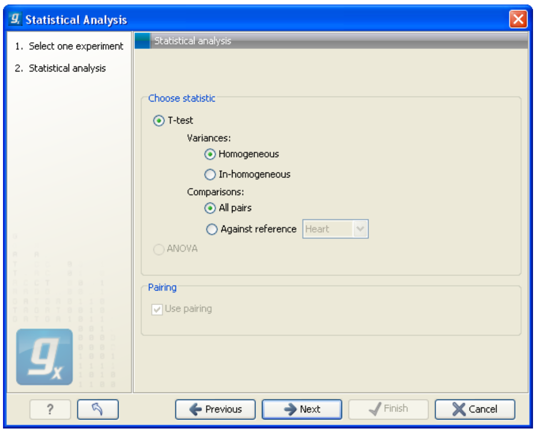
20.4.1 Gaussian-based tests

The tests based on the Gaussian distribution essentially compare the mean expression level in the experimental groups in the study, and evaluates the significance of the difference relative to the variance (or 'spread') of the data within the groups. The details of the formula used for calculating the test statistics vary according to the experimental setup and the assumptions you make about the data (read more about this in the sections on t-test and ANOVA below). The explanation of how to proceed is divided into two, depending on how many groups there are in your experiment. First comes the explanation for t-tests which is the only analysis available for two-group experimental setups (t-tests can also be used for pairwise comparison of groups in multi-group experiments). Next comes an explanation of the ANOVA test which can be used for multi-group experiments.

Note that the test statistics for the t-test and ANOVA analysis use the estimated group variances in their denominators. If all expression values in a group are identical the estimated variance for that group will be zero. If the estimated variances for both (or all) groups are zero the denominator of the test statistic will be zero. The numerator's value depends on the difference of the group means. If this is zero, the numerator is zero and the test statistic will be 0/0 which is NaN. If the numerator is different from zero the test statistic will be + or - infinity, depending on which group mean is bigger. If all values in all groups are identical the test statistic is set to zero.

T-tests

For experiments with two groups you can, among the Gaussian tests, only choose a **T-test** as shown in figure 20.38.



Choose statistic

● T-test

Variances:
● Homogeneous
○ In-homogeneous

Comparisons:
● All pairs
○ Against reference Heart

○ ANOVA

Pairing

☑ Use pairing

Figure 20.38: *Selecting a t-test.*

There are different types of t-tests, depending on the assumption you make about the variances in the groups. By selecting 'Homogeneous' (the default) calculations are done assuming that the groups have equal variances. When 'In-homogeneous' is selected, this assumption is not made.

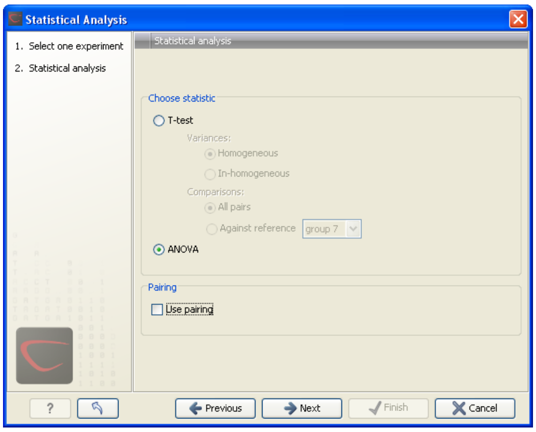
The t-test can also be chosen if you have a multi-group experiment. In this case you may choose either to have t-tests produced for all pairs of groups (by clicking the 'All pairs' button) or to have a t-test produced for each group compared to a specified reference group (by clicking the 'Against reference' button). In the last case you must specify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

If a experiment with pairing was set up (see section 20.1.2) the **Use pairing** tick box is active. If ticked, paired t-tests will be calculated, if not, the formula for the standard t-test will be used.

When a t-test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Difference' column contains the difference between the mean of the expression values across the samples assigned to group 2 and the mean of the expression values across the samples assigned to group 1. The 'Fold Change' column tells you how many times bigger the mean expression value in group 2 is relative to that of group 1. If the mean expression value in group 2 is bigger than that in group 1 this value is the mean expression value in group 2 divided by that in group 1. If the mean expression value in group 2 is smaller than that in group 1 the fold change is the mean expression value in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 20.4.3).

ANOVA

For experiments with more than two groups you can choose **T-test** as described above, or **ANOVA** as shown in figure 20.39.



Statistical Analysis

1. Select one experiment
2. Statistical analysis

Statistical analysis

Choose statistic

○ T-test

Variances:

● Homogeneous

○ In-homogeneous

Comparisons:

● All pairs

○ Against reference group 7

● ANOVA

Pairing

□ Use pairing

Figure 20.39: *Selecting ANOVA.*

The ANOVA method allows analysis of an experiment with one factor and a number of groups, e.g. different types of tissues, or time points. In the analysis, the variance within groups is compared to the variance between groups. You get a significant result (that is, a small ANOVA p-value) if the difference you see between groups relative to that within groups, is larger than what you would expect, if the data were really drawn from groups with equal means.

If an experiment with pairing was set up (see section 20.1.2) the **Use pairing** tick box is active.

If ticked, a repeated measures one-way ANOVA test will be calculated, if not, the formula for the standard one-way ANOVA will be used.

When an ANOVA analysis is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Max difference' column contains the difference between the maximum and minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs after the group with the minimum mean expression value (with the ordering: group 1, group 2, ...). The 'Max fold change' column contains the ratio of the maximum of the mean expression values of the groups to the minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs after the group with the minimum mean expression value (with the ordering: group 1, group 2, ...). The 'Test statistic' column holds the value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 20.4.3).

20.4.2 Tests on proportions

The proportions-based tests are applicable in situations where your data samples consists of counts of a number of 'types' of data. This could e.g. be in a study where gene expression levels are measured by RNA-Seq or tag profiling. Here the different 'types' could correspond to the different 'genes' in a reference genome, and the counts could be the numbers of reads matching each of these genes. The tests compare counts by considering the proportions that they make up the total sum of counts in each sample. By comparing the expression levels at the level of proportions rather than raw counts, the data is corrected for sample size.

There are two tests available for comparing proportions: the test of [Kal et al., 1999] and the test of [Baggerly et al., 2003]. Both tests compare pairs of groups. If you have a multi-group experiment (see section 20.1.2), you may choose either to have tests produced for all pairs of groups (by clicking the 'All pairs' button) or to have a test produced for each group compared to a specified reference group (by clicking the 'Against reference' button). In the last case you must specify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

Note that the proportion-based tests use the total sample counts (that is, the sum over all expression values). If one (or more) of the counts are NaN, the sum will be NaN and all the test statistics will be NaN. As a consequence all p-values will also be NaN. You can avoid this by filtering your experiment and creating a new experiment so that no NaN values are present, before you apply the tests.

Kal et al.'s test (Z-test)

Kal et al.'s test [Kal et al., 1999] compares a single sample against another single sample, and thus requires that each group in you experiment has only one sample. The test relies on an approximation of the binomial distribution by the normal distribution [Kal et al., 1999]. Considering proportions rather than raw counts the test is also suitable in situations where the sum of counts is different between the samples.

When Kal's test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Proportions difference' column contains the difference between the proportion in group 2 and the proportion in group 1. The 'Fold Change' column tells you how many times bigger the proportion in group 2 is relative to that of group 1. If the proportion in group 2 is bigger than that in group 1 this value is the proportion in group 2 divided by that in group 1. If the proportion in group 2 is smaller than that in group 1 the fold change is the proportion in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 20.4.3).

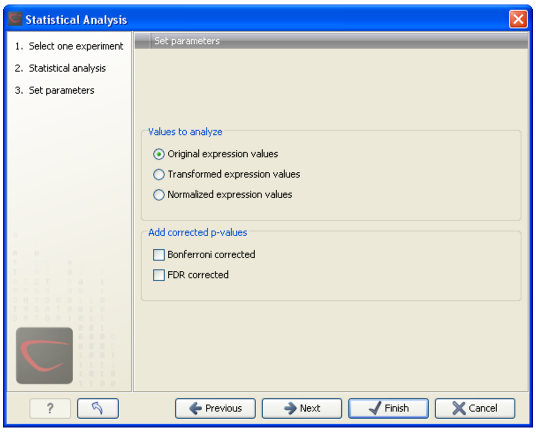
Baggerley et al.'s test (Beta-binomial)

Baggerley et al.'s test [Baggerly et al., 2003] compares the proportions of counts in a group of samples against those of another group of samples, and is suited to cases where replicates are available in the groups. The samples are given different weights depending on their sizes (total counts). The weights are obtained by assuming a Beta distribution on the proportions in a group, and estimating these, along with the proportion of a binomial distribution, by the method of moments. The result is a weighted t-type test statistic.

When Baggerley's test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Weighted proportions difference' column contains the difference between the mean of the weighted proportions across the samples assigned to group 2 and the mean of the weighted proportions across the samples assigned to group 1. The 'Fold Change' column tells you how many times bigger the mean of the weighted proportions in group 2 is relative to that of group 1. If the mean of the weighted proportions in group 2 is bigger than that in group 1 this value is the mean of the weighted proportions in group 2 divided by that in group 1. If the mean of the weighted proportions in group 2 is smaller than that in group 1 the fold change is the mean of the weighted proportions in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 20.4.3).

20.4.3 Corrected p-values

Clicking **Next** will display a dialog as shown in figure 20.40.



Statistical Analysis

1. Select one experiment
2. Statistical analysis
3. Set parameters

Set parameters

Values to analyze

● Original expression values
○ Transformed expression values
○ Normalized expression values

Add corrected p-values

□ Bonferroni corrected
□ FDR corrected

Figure 20.40: *Additional settings for the statistical analysis.*

At the top, you can select which values to analyze (see section 20.2.1).

Below you can select to add two kinds of corrected p-values to the analysis (in addition to the standard p-value produced for the test statistic):

- **Bonferroni corrected**.
- **FDR corrected**.

Both are calculated from the original p-values, and aim in different ways to take into account the issue of multiple testing [Dudoit et al., 2003]. The problem of multiple testing arises because the original p-values are related to a single test: the p-value is the probability of observing a more extreme value than that observed in the test carried out. If the p-value is 0.04, we would expect an as extreme value as that observed in 4 out of 100 tests carried out among groups with no difference in means. Popularly speaking, if we carry out 10000 tests and select the features with original p-values below 0.05, we will expect about 0.05 times 10000 = 500 to be false positives.

The Bonferroni corrected p-values handle the multiple testing problem by controlling the 'family-wise error rate': the probability of making at least one false positive call. They are calculated by multiplying the original p-values by the number of tests performed. The probability of having at least one false positive among the set of features with Bonferroni corrected p-values below 0.05, is less than 5%. The Bonferroni correction is conservative: there may be many genes that are differentially expressed among the genes with Bonferroni corrected p-values above 0.05, that will be missed if this correction is applied.

Instead of controlling the family-wise error rate we can control the false discovery rate: FDR. The false discovery rate is the proportion of false positives among all those declared positive. We expect 5 % of the features with FDR corrected p-values below 0.05 to be false positive. There are many methods for controlling the FDR - the method used in *CLC Genomics Workbench* is that of [Benjamini and Hochberg, 1995].

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Note that if you have already performed statistical analysis on the same values, the existing one will be overwritten.

20.4.4 Volcano plots - inspecting the result of the statistical analysis

The results of the statistical analysis are added to the experiment and can be shown in the experiment table (see section 20.1.3). Typically columns containing the differences (or weighted differences) of the mean group values and the fold changes (or weighted fold changes) of the mean group values will be added along with a column of p-values. Also, columns with FDR or Bonferroni corrected p-values will be added if these were calculated. This added information allows features to be sorted and filtered to exclude the ones without sufficient proof of differential expression (learn more in section C).

If you want a more visual approach to the results of the statistical analysis, you can click the **Show Volcano Plot** () button at the bottom of the experiment table view. In the same way as the scatter plot presented in section 20.1.5, the volcano plot is yet another view on the experiment. Because it uses the p-values and mean differences produced by the statistical analysis, the plot is only available once a statistical analysis has been performed on the experiment.

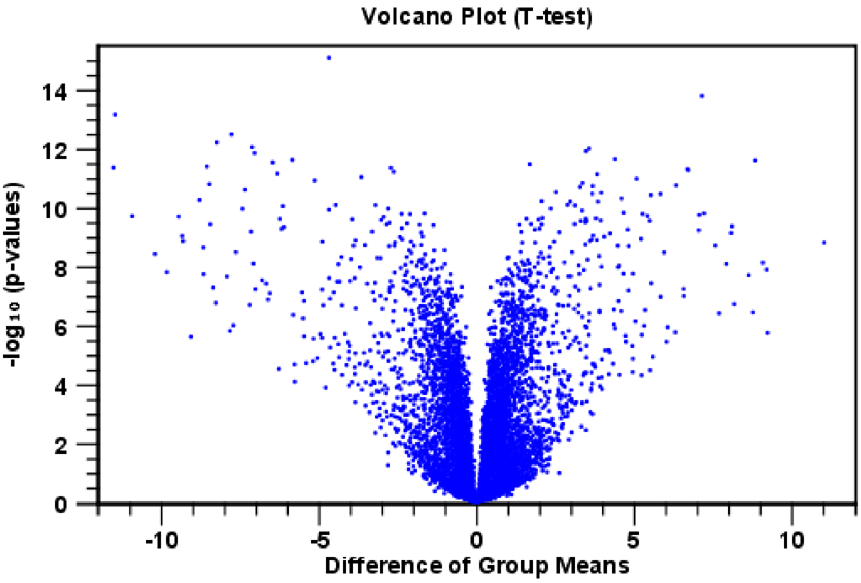


Figure 20.41: *Volcano plot.*

An example of a volcano plot is shown in figure 20.41.

The volcano plot shows the relationship between the p-values of a statistical test and the magnitude of the difference in expression values of the samples in the groups. On the y-axis the $-\log_{10}$ p-values are plotted. For the x-axis you may choose between two sets of values by choosing either 'Fold change' or 'Difference' in the volcano plot side panel's 'Values' part. If you choose 'Fold change' the log of the values in the 'fold change' (or 'Weighted fold change') column for the test will be displayed. If you choose 'Difference' the values in the 'Difference' (or 'Weighted difference') column will be used. Which values you wish to display will depend upon the scale of you data (Read the note on fold change in section 20.1.3).

The larger the difference in expression of a feature, the more extreme it's point will lie on the X-axis. The more significant the difference, the smaller the p-value and thus the higher the $-\log_{10}(p)$ value. Thus, points for features with highly significant differences will lie high in the plot. Features of interest are typically those which change significantly and by a certain magnitude. These are the points in the upper left and upper right hand parts of the volcano plot.

If you have performed different tests or you have an experiment with multiple groups you need to specify for which test and which group comparison you want the volcano plot to be shown. You do this in the 'Test' and 'Values' parts of the volcano plot side panel.

Options for the volcano plot are described in further detail when describing the **Side Panel** below.

If you place your mouse on one of the dots, a small text box will tell the name of the feature. Note that you can zoom in and out on the plot (see section 3.3).

In the **Side Panel** to the right, there is a number of options to adjust the view of the volcano plot. Under **Graph preferences**, you can adjust the general properties of the volcano plot

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.

- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

Below the general preferences, you find the **Dot properties**, where you can adjust coloring and appearance of the dots.

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

At the very bottom, you find two groups for choosing which values to display:

- **Test**. In this group, you can select which kind of test you want the volcano plot to be shown for.
- **Values**. Under **Values**, you can select which values to plot. If you have multi-group experiments, you can select which groups to compare. You can also select whether to plot **Difference** or **Fold change** on the x-axis. Read the note on fold change in section 20.1.3.

Note that if you wish to use the same settings next time you open a box plot, you need to save the settings of the **Side Panel** (see section 5.6).

20.5 Feature clustering

Feature clustering is used to identify and cluster together features with similar expression patterns over samples (or experimental groups). Features that cluster together may be involved in the same biological process or be co-regulated. Also, by examining annotations of genes within a cluster, one may learn about the underlying biological processes involved in the experiment studied.

20.5.1 Hierarchical clustering of features

A hierarchical clustering of features is a tree presentation of the similarity in expression profiles of the features over a set of samples (or groups). The tree structure is generated by

1. letting each feature be a cluster
2. calculating pairwise distances between all clusters
3. joining the two closest clusters into one new cluster
4. iterating 2-3 until there is only one cluster left (which will contain all samples).

The tree is drawn so that the distances between clusters are reflected by the lengths of the branches in the tree. Thus, features with expression profiles that closely resemble each other have short distances between them, those that are more different, are placed further apart.

To start the clustering of features:

> **Toolbox** | **Expression Analysis ( )** | **Feature Clustering** | **Hierarchical Clustering of Features ( )**

Select at least two samples (( ) or ( )) or an experiment ( ).

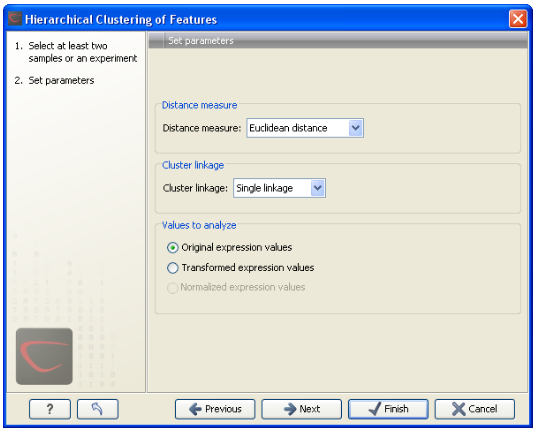
Note! If your data contains many features, the clustering will take very long time and could make your computer unresponsive. It is recommended to perform this analysis on a subset of the data (which also makes it easier to make sense of the clustering. Typically, you will want to filter away the features that are thought to represent only noise, e.g. those with mostly low values, or with little difference between the samples). See how to create a sub-experiment in section 20.1.3.

Clicking **Next** will display a dialog as shown in figure 20.42. The hierarchical clustering algorithm requires that you specify a distance measure and a cluster linkage. The distance measure is used specify how distances between two features should be calculated. The cluster linkage specifies how you want the distance between two clusters, each consisting of a number of features, to be calculated.

At the top, you can choose three kinds of **Distance measures**:

- **Euclidean distance**. The ordinary distance between two points - the length of the segment connecting them. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean distance between u and v is

$$|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.$$


Hierarchical Clustering of Features
1. Select at least two samples or an experiment
2. Set parameters
Set parameters
?
Previous
Next
Finish
Cancel

Distance measure

Distance measure: Euclidean distance

Cluster linkage

Cluster linkage: Single linkage

Values to analyze

● Original expression values

○ Transformed expression values

○ Normalized expression values

Figure 20.42: *Parameters for hierarchical clustering of features.*

- **1 - Pearson correlation.** The Pearson correlation coefficient between two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ is defined as

$$r = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x_i - \overline{x}}{s_x}) * (\frac{y_i - \overline{y}}{s_y})$$

  where $\overline{x}/\overline{y}$ is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance**. The Manhattan distance between two points is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between u and v is

$$|u - v| = \sum_{i=1}^{n}|u_i - v_i|.$$

Next, you can select different ways to calculate distances between clusters. The possible cluster linkage to use are:

- **Single linkage**. The distance between two clusters is computed as the distance between the two closest elements in the two clusters.

- **Average linkage**. The distance between two clusters is computed as the average distance between objects from the first cluster and objects from the second cluster. The averaging is performed over all pairs (x, y), where x is an object from the first cluster and y is an object from the second cluster.

- **Complete linkage**. The distance between two clusters is computed as the maximal object-to-object distance $d(x_i, y_j)$, where x_i comes from the first cluster, and y_j comes from the second cluster. In other words, the distance between two clusters is computed as the distance between the two farthest objects in the two clusters.

At the bottom, you can select which values to cluster (see section 20.2.1). Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Result of hierarchical clustering of features

The result of a feature clustering is shown in figure 20.43.

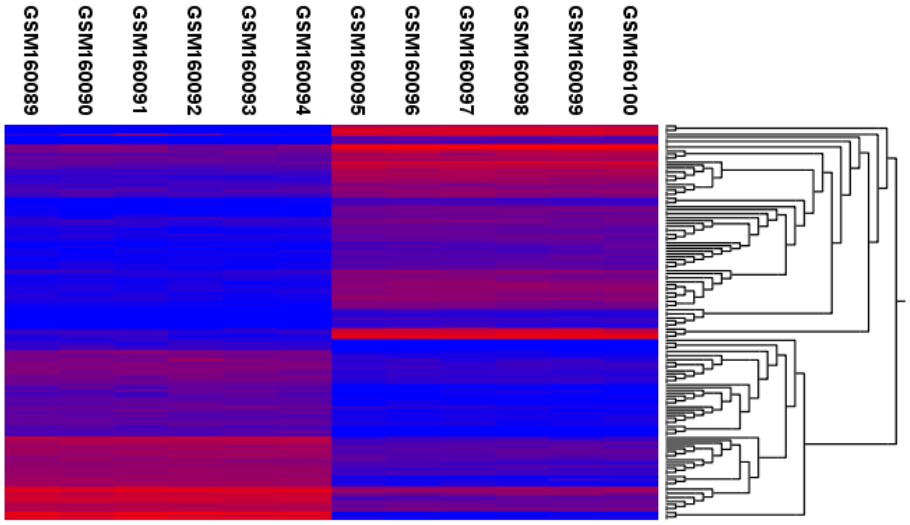


Figure 20.43: *Hierarchical clustering of features.*

If you have used an **experiment** (▦) as input, the clustering is added to the experiment and will be saved when you save the experiment. It can be viewed by clicking the **Show Heat Map** (▦) button at the bottom of the view (see figure 20.44).

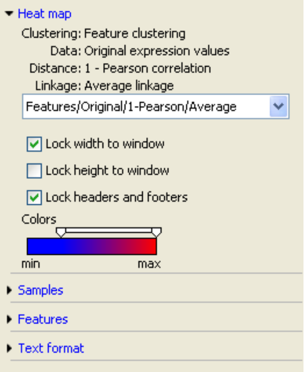
Figure 20.44: *Showing the hierarchical clustering of an experiment.*

If you have selected a number of **samples** ((▦) or (▤)) as input, a new element will be created that has to be saved separately.

Regardless of the input, a hierarchical tree view with associated heatmap is produced (figure 20.43). In the heatmap each row corresponds to a feature and each column to a sample. The color in the i'th row and j'th column reflects the expression level of feature i in sample j (the color scale can be set in the side panel). The order of the rows in the heatmap are determined by the hierarchical clustering. If you place the mouse on one of the rows, you will see the name of the corresponding feature to the left. The order of the columns (that is, samples) is determined by their input order or (if defined) experimental grouping. The names of the samples are listed at the top of the heatmap and the samples are organized into groups.

There are a number of options to change the appearance of the heat map. At the top of the **Side Panel**, you find the **Heat map** preference group (see figure 20.45).

At the top, there is information about the heat map currently displayed. The information regards type of clustering, expression value used together with distance and linkage information. If you have performed more than one clustering, you can choose between the resulting heat maps in a drop-down box (see figure 20.46).



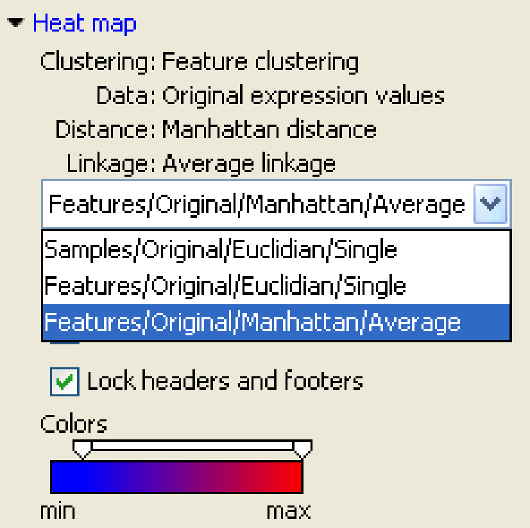
Heat map

Clustering: Feature clustering
Data: Original expression values
Distance: 1 - Pearson correlation
Linkage: Average linkage

Features/Original/1-Pearson/Average

☑ Lock width to window
☐ Lock height to window
☑ Lock headers and footers

Figure 20.45: *Side Panel of heat map.*



Heat map

Clustering: Feature clustering
Data: Original expression values
Distance: Manhattan distance
Linkage: Average linkage

Features/Original/Manhattan/Average

Samples/Original/Euclidian/Single
Features/Original/Euclidian/Single
Features/Original/Manhattan/Average

☑ Lock headers and footers

Figure 20.46: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

Note that if you perform an identical clustering, the existing heat map will simply be replaced. Below this box, there is a number of settings for displaying the heat map.

- **Lock width to window**. When you zoom in the heat map, you will per default only zoom in on the vertical level. This is because the width of the heat map is locked to the window. If you uncheck this option, you will zoom both vertically and horizontally. Since you always have more features than samples, it is useful to lock the width since you then have all the samples in view all the time.

- **Lock height to window**. This is the corresponding option for the height. Note that if you check both options, you will not be able to zoom at all, since both the width and the height is fixed.

- **Lock headers and footers**. This will ensure that you are always able to see the sample and feature names and the trees when you zoom in.

- **Colors**. The expression levels are visualized using a gradient color scheme, where the right side color is used for high expression levels and the left side color is used for low expression levels. You can change the coloring by clicking the box, and you can change the relative coloring of the values by dragging the two knobs on the white slider above.

Below you find the **Samples** and **Features** groups. They contain options to show names above/below and left/right, respectively. Furthermore, they contain options to show the tree above/below or left/right, respectively. Note that for clustering of samples, you find the tree options in the **Samples** group, and for clustering of features, you find the tree options in the **Features** group. With the tree options, you can also control the **Tree size**, from tiny to very large, and the option of showing the full tree, no matter how much space it will use.

Note that if you wish to use the same settings next time you open a heat map, you need to save the settings of the **Side Panel** (see section 5.6).

20.5.2 K-means/medoids clustering

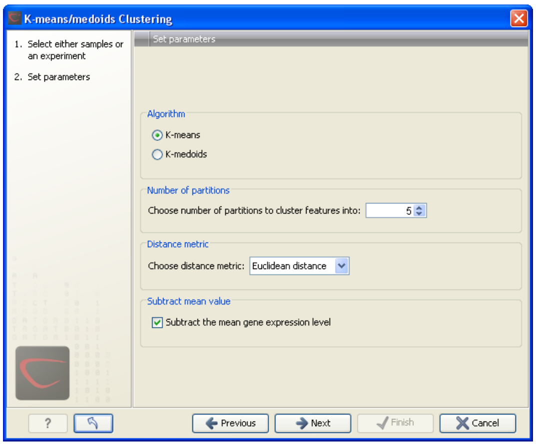
In a k-means or medoids clustering, features are clustered into k separate clusters. The procedures seek to find an assignment of features to clusters, for which the distances between features within the cluster is small, while distances between clusters are large.

> **Toolbox | Expression Analysis () | Feature Clustering | K-means/medoids Clustering ()**

Select at least two samples (() or ()) or an experiment ().

Note! If your data contains many features, the clustering will take very long time and could make your computer unresponsive. It is recommended to perform this analysis on a subset of the data (which also makes it easier to make sense of the clustering). See how to create a sub-experiment in section 20.1.3.

Clicking **Next** will display a dialog as shown in figure 20.47.



1. Select either samples or an experiment
2. Set parameters

Algorithm
● K-means
○ K-medoids

Number of partitions
Choose number of partitions to cluster features into: 5

Distance metric
Choose distance metric: Euclidean distance

Subtract mean value
☑ Subtract the mean gene expression level

Figure 20.47: *Parameters for k-means/medoids clustering.*

The parameters are:

- **Algorithm**. You can choose between two clustering methods:
  - **K-means**. K-means clustering assigns each point to the cluster whose center is nearest. The center/centroid of a cluster is defined as the average of all points in the cluster. If a data set has three dimensions and the cluster has two points $X = (x_1, x_2, x_3)$ and $Y = (y_1, y_2, y_3)$, then the centroid Z becomes $Z = (z_1, z_2, z_3)$, where $z_i = (x_i + y_i)/2$ for $i = 1, 2, 3$. The algorithm attempts to minimize the

intra-cluster variance defined by:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

where there are k clusters $S_i, i = 1, 2, \ldots, k$ and μ_i is the centroid of all points $x_j \in S_i$. The detailed algorithm can be found in [Lloyd, 1982].

- **K-medoids**. K-medoids clustering is computed using the PAM-algorithm (PAM is short for Partitioning Around Medoids). It chooses datapoints as centers in contrast to the K-means algorithm. The PAM-algorithm is based on the search for k representatives (called medoids) among all elements of the dataset. When having found k representatives k clusters are now generated by assigning each element to its nearest medoid. The algorithm first looks for a good initial set of medoids (the BUILD phase). Then it finds a local minimum for the objective function:

  $$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - c_i)^2$$

  where there are k clusters $S_i, i = 1, 2, \ldots, k$ and c_i is the medoid of S_i. This solution implies that there is no single switch of an object with a medoid that will decrease the objective (this is called the SWAP phase). The PAM-agorithm is described in [Kaufman and Rousseeuw, 1990].

- **Number of partitions**. The number of partitions to cluster features into.

- **Distance metric**. The metric to compute distance between data points.

  - **Euclidean distance**. The ordinary distance between two elements - the length of the segment connecting them. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean distance between u and v is

    $$|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.$$

  - **Manhattan distance**. The Manhattan distance between two elements is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between u and v is

    $$|u - v| = \sum_{i=1}^{n} |u_i - v_i|.$$

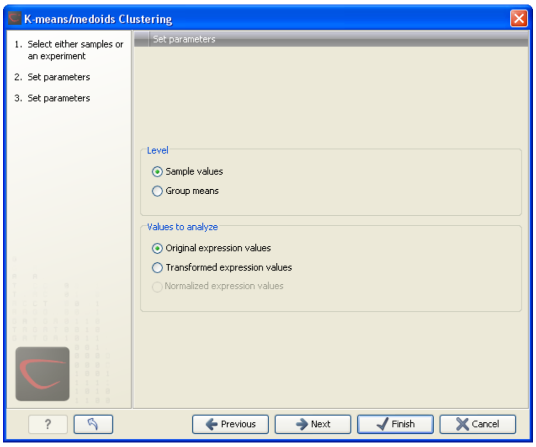
- **Subtract mean value**. For each gene, subtract the mean gene expression value over all input samples.

Clicking **Next** will display a dialog as shown in figure 20.48.

At the top, you can choose the **Level** to use. Choosing 'sample values' means that distances will be calculated using all the individual values of the samples. When 'group means' are chosen, distances are calculated using the group means.

At the bottom, you can select which values to cluster (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.



Level

● Sample values
○ Group means

Values to analyze

● Original expression values
○ Transformed expression values
○ Normalized expression values

Figure 20.48: *Parameters for k-means/medoids clustering.*

Viewing the result of k-means/medoids clustering

The result of the clustering is a number of graphs. The number depends on the number of partitions chosen (figure 20.47) - there is one graph per cluster. Using drag and drop as explained in section 3.2.6, you can arrange the views to see more than one graph at the time.

Figure 20.49 shows an example where four clusters have been arranged side-by-side.

The samples used are from a time-series experiment, and you can see that the expression levels for each cluster have a distinct pattern. The two clusters at the bottom have falling and rising expression levels, respectively, and the two clusters at the top both fall at the beginning but then rise again (the one to the right starts to rise earlier that the other one).

Having inspected the graphs, you may wish to take a closer look at the features represented in each cluster. In the experiment table, the clustering has added an extra column with the name of the cluster that the feature belongs to. In this way you can filter the table to see only features from a specific cluster. This also means that you can select the feature of this cluster in a volcano or scatter plot as described in section 20.1.6.

20.6 Annotation tests

The annotation tests are tools for detecting significant patterns among features (e.g. genes) of experiments, based on their annotations. This may help in interpreting the analysis of the large numbers of features in an experiment in a biological context. Which biological context, depends on which annotation you choose to examine, and could e.g. be biological process, molecular function or pathway as specified by the Gene Ontology or KEGG. The annotation testing tools of course require that the features in the experiment you want to analyze are annotated. Learn how to annotate an experiment in section 20.1.4.

20.6.1 Hypergeometric tests on annotations

The first approach to using annotations to extract biological information is the hypergeometric annotation test. This test measures the extend to which the annotation categories of features in a smaller gene list, 'A', are over or under-represented relative to those of the features in larger gene list 'B', of which 'A' is a sub-list. Gene list B is often the features of the full experiment, possibly with features which are thought to represent only noise, filtered away. Gene list A is

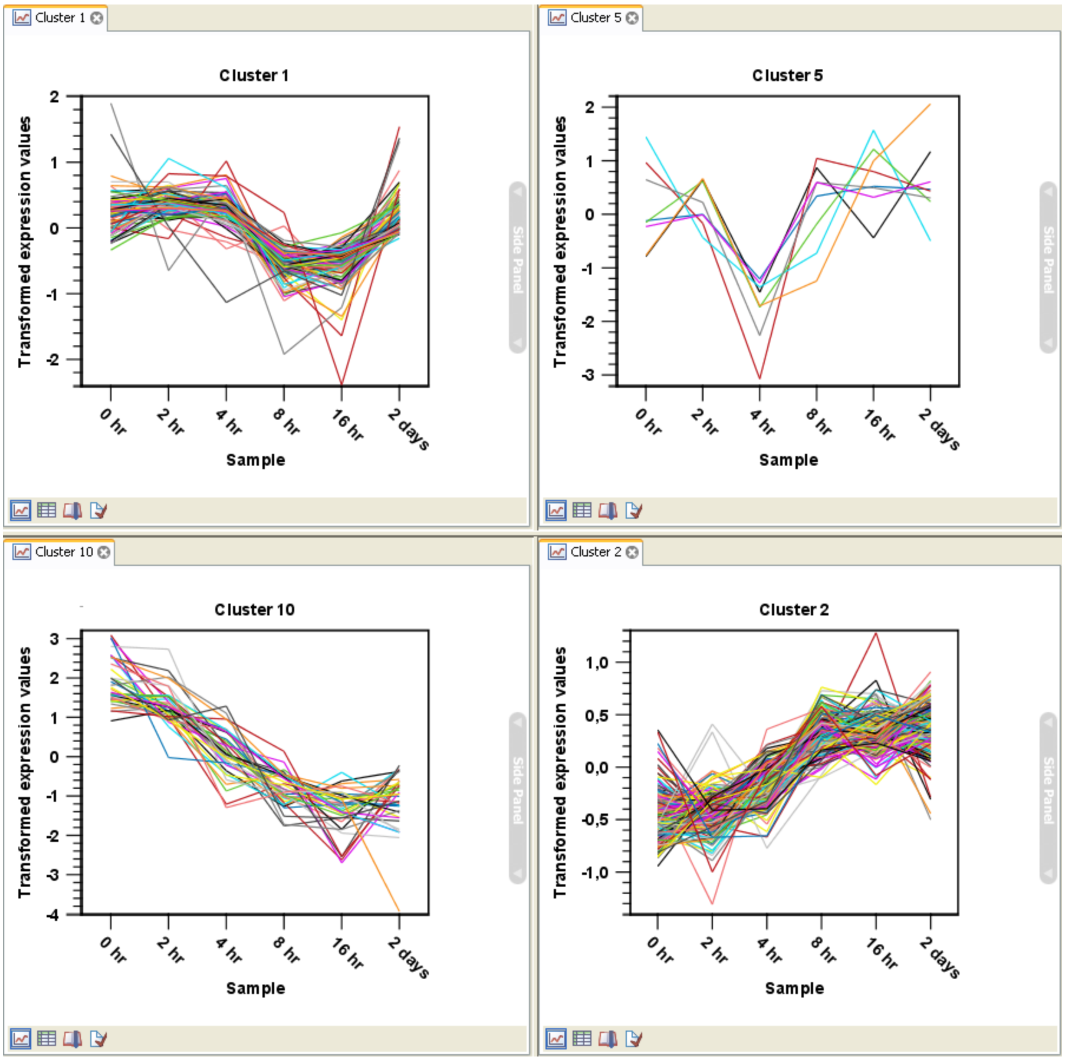


Figure 20.49: *Four clusters created by k-means/medoids clustering.*

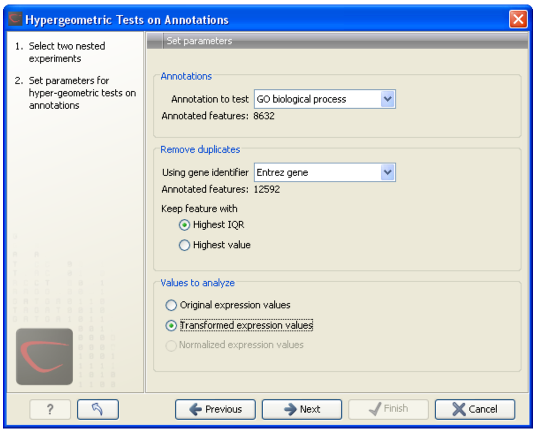
a sub-experiment of the full experiment where most features have been filtered away and only those that seem of interest are kept. Typically gene list A will consist of a list of candidate differentially expressed genes. This could be the gene list obtained after carrying out a statistical analysis on the experiment, and keeping only features with FDR corrected p-values <0.05 and a fold change which is larger than 2 in absolute value. The hyper geometric test procedure implemented is similar to the unconditional GOstats test of [Falcon and Gentleman, 2007].

> **Toolbox | Expression Analysis ( ) | Annotation Test | Hypergeometric Tests on Annotations ( )**

This will show a dialog where you can select the two experiments - the larger experiment, e.g. the original experiment including the full list of features - and a sub-experiment (see how to create a sub-experiment in section 20.1.3).

Click **Next**. This will display the dialog shown in figure 20.50.

At the top, you select which annotation to use for testing. You can select from all the annotations available on the experiment, but it is of course only a few that are biologically relevant. Once you have selected an annotation, you will see the number of features carrying this annotation below.



Hypergeometric Tests on Annotations

1. Select two nested experiments
2. Set parameters for hyper-geometric tests on annotations

Set parameters

Annotations

Annotation to test: GO biological process

Annotated features: 8632

Remove duplicates

Using gene identifier: Entrez gene

Annotated features: 12592

Keep feature with

● Highest IQR

○ Highest value

Values to analyze

○ Original expression values

● Transformed expression values

○ Normalized expression values

Figure 20.50: *Parameters for performing a hypergeometric test on annotations*

Annotations are typically given at the gene level. Often a gene is represented by more than one feature in an experiment. If this is not taken into account it may lead to a biased result. The standard way to deal with this is to reduce the set of features considered, so that each gene is represented only once. In the next step, **Remove duplicates**, you can choose how you want this to be done:

- **Using gene identifier**.
- **Keep feature with**:
  - **Highest IQR**. The feature with the highest interquartile range (IQR) is kept.
  - **Highest value**. The feature with the highest expression value is kept.

First you specify which annotation you want to use as gene identifier. Once you have selected this, you will see the number of features carrying this annotation below. Next you specify which feature you want to keep for each gene. This may be either the feature with the highest inter-quartile range or the highest value.

At the bottom, you can select which values to analyze (see section 20.2.1).

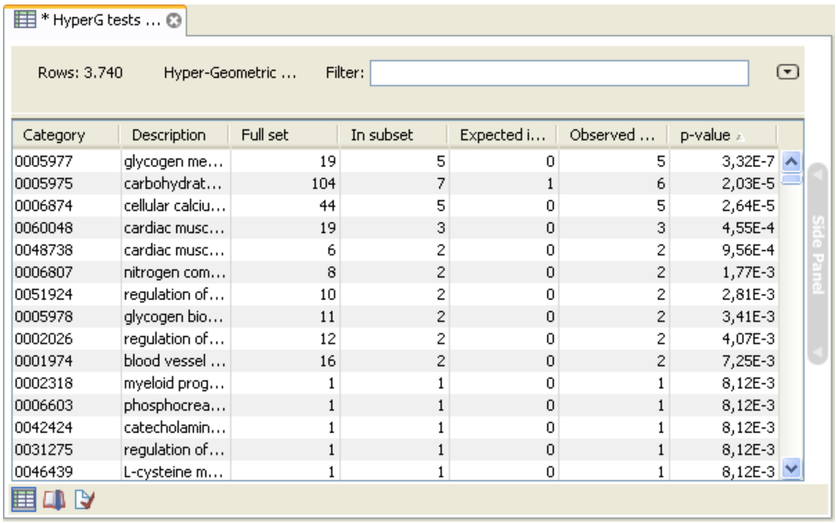
Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Result of hypergeometric tests on annotations

The result of performing hypergeometric tests on annotations using GO biological process is shown in figure 20.51.

The table shows the following information:

- **Category**. This is the identifier for the category.
- **Description**. This is the description belonging to the category. Both of these are simply extracted from the annotations.
- **Full set**. The number of features in the original experiment (not the subset) with this category. (Note that this is after removal of duplicates).



Rows: 3.740 Hyper-Geometric ... Filter:

Category	Description	Full set	In subset	Expected i...	Observed ...	p-value
0005977	glycogen me...	19	5	0	5	3,32E-7
0005975	carbohydrat...	104	7	1	6	2,03E-5
0006874	cellular calciu...	44	5	0	5	2,64E-5
0060048	cardiac musc...	19	3	0	3	4,55E-4
0048738	cardiac musc...	6	2	0	2	9,56E-4
0006807	nitrogen com...	8	2	0	2	1,77E-3
0051924	regulation of...	10	2	0	2	2,81E-3
0005978	glycogen bio...	11	2	0	2	3,41E-3
0002026	regulation of...	12	2	0	2	4,07E-3
0001974	blood vessel ...	16	2	0	2	7,25E-3
0002318	myeloid prog...	1	1	0	1	8,12E-3
0006603	phosphocrea...	1	1	0	1	8,12E-3
0042424	catecholamin...	1	1	0	1	8,12E-3
0031275	regulation of...	1	1	0	1	8,12E-3
0046439	L-cysteine m...	1	1	0	1	8,12E-3

Figure 20.51: *The result of testing on GO biological process.*

- **In subset**. The number of features in the the subset with this category. (Note that this is after removal of duplicates).
- **Expected in subset**. The number of features we would have expected to find with this annotation category in the subset, if the subset was a random draw from the full set.
- **Observed - expected**. 'In subset' - 'Expected in subset'
- **p-value**. The tail probability of the hyper geometric distribution This is the value used for sorting the table.

Categories with small p-values are categories that are over or under-represented on the features in the subset relative to the full set.

20.6.2 Gene set enrichment analysis

When carrying out a hypergeometric test on annotations you typically compare the annotations of the genes in a subset containing 'the significantly differentially expressed genes' to those of the total set of genes in the experiment. Which, and how many, genes are included in the subset is somewhat arbitrary - using a larger or smaller p-value cut-off will result in including more or less. Also, the magnitudes of differential expression of the genes is not considered.

The Gene Set Enrichment Analysis (GSEA) does NOT take a sublist of differentially expressed genes and compare it to the full list - it takes a single gene list (a single experiment). The idea behind GSEA is to consider a measure of association between the genes and phenotype of interest (e.g. test statistic for differential expression) and rank the genes according to this measure of association. A test is then carried out for each annotation category, for whether the ranks of the genes in the category are evenly spread throughout the ranked list, or tend to occur at the top or bottom of the list.

The GSEA test implemented here is that of [Tian et al., 2005]. The test implicitly calculates and uses a standard t-test statistic for two-group experiments, and ANOVA statistic for multiple group experiments for each feature, as measures of association. For each category, the test statistics for the features in than category are summed and a category based test statistic is calculated as this sum divided by the square root of the number of features in the category. Note that if a feature has the value NaN in one of the samples, the t-test statistic for the feature will be NaN. Consequently, the combined statistic for each of the categories in which the feature is included will be NaN. Thus, it is advisable to filter out any feature that has a NaN value before applying GSEA.

The p-values for the GSEA test statistics are calculated by permutation: The original test statistics for the features are permuted and new test statistics are calculated for each category, based on the permuted feature test statistics. This is done the number of times specified by the user in the wizard. For each category, the lower and upper tail probabilities are calculated by comparing the original category test statistics to the distribution of the permutation-based test statistics for that category. The lower and higher tail probabilities are the number of these that are lower and higher, respectively, than the observed value, divided by the number of permutations.

As the p-values are based on permutations you may some times see results where category x's test statistic is lower than that of category y and the categories are of equal size, but where the lower tail probability of category x is higher than that of category y. This is due to imprecision in the estimations of the tail probabilities from the permutations. The higher the number of permutations, the more stable the estimation.

You may run a GSEA on a full experiment, or on a sub-experiment where you have filtered away features that you think are un-informative and represent only noise. Typically you will remove features that are constant across samples (those for which the value in the 'Range' column is zero' — these will have a t-test statistic of zero) and/or those for which the inter-quantile range is small. As the GSEA algorithm calculates and ranks genes on p-values from a test of differential expression, it will generally not make sense to filter the experiment on p-values produced in an analysis if differential expression, prior to running GSEA on it.

> **Toolbox | Expression Analysis ( ) | Annotation Test | Gene Set Enrichment Analysis (GSEA) ( )**

Select an experiment and click **Next**.

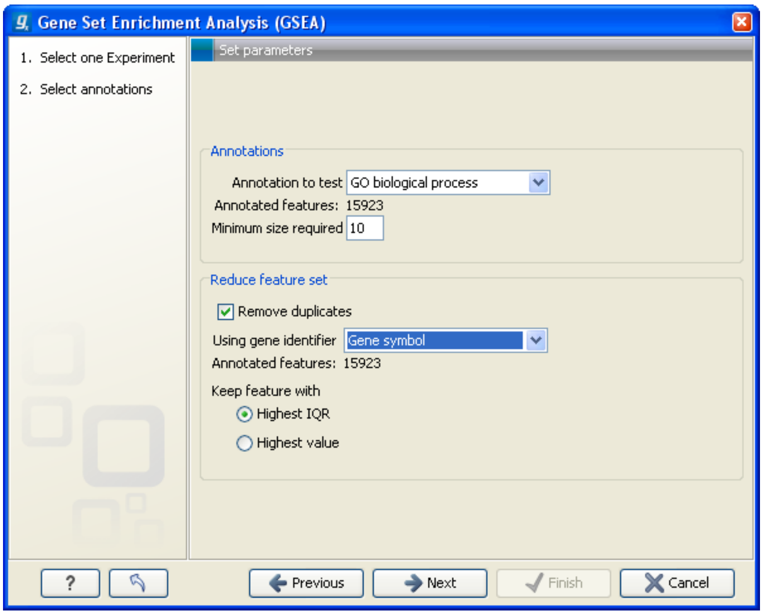
Click **Next**. This will display the dialog shown in figure 20.52.

At the top, you select which annotation to use for testing. You can select from all the annotations available on the experiment, but it is of course only a few that are biologically relevant. Once you have selected an annotation, you will see the number of features carrying this annotation below.

In addition, you can set a filter: **Minimum size required**. Only categories with more genes (i.e. features) than the specified number will be considered. Excluding categories with small numbers of genes may lead to more robust results.

Annotations are typically given at the gene level. Often a gene is represented by more than one feature in an experiment. If this is not taken into account it may lead to a biased result. The standard way to deal with this is to reduce the set of features considered, so that each gene is represented only once. Check the **Remove duplicates** check box to reduce the feature set, and you can choose how you want this to be done:

- **Using gene identifier**.
- **Keep feature with**:
  - **Highest IQR**. The feature with the highest interquartile range (IQR) is kept.



Gene Set Enrichment Analysis (GSEA)

1. Select one Experiment
2. Select annotations

Set parameters

Annotations

Annotation to test: GO biological process

Annotated features: 15923

Minimum size required: 10

Reduce feature set

☑ Remove duplicates

Using gene identifier: Gene symbol

Annotated features: 15923

Keep feature with

● Highest IQR

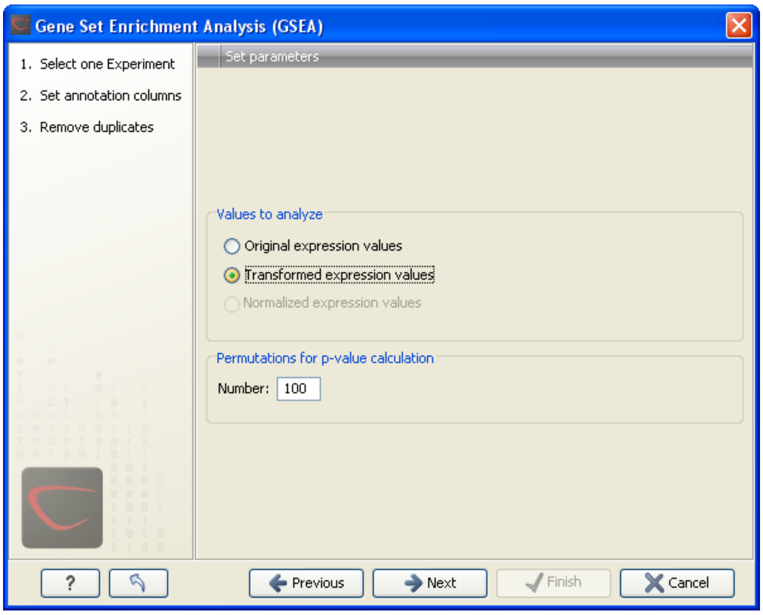
○ Highest value

Figure 20.52: *Gene set enrichment analysis on GO biological process*

- **Highest value**. The feature with the highest expression value is kept.

First you specify which annotation you want to use as gene identifier. Once you have selected this, you will see the number of features carrying this annotation below. Next you specify which feature you want to keep for each gene. This may be either the feature with the highest inter-quartile range or the highest value.

Clicking **Next** will display the dialog shown in figure 20.53.



Gene Set Enrichment Analysis (GSEA)

1. Select one Experiment
2. Set annotation columns
3. Remove duplicates

Set parameters

Values to analyze

○ Original expression values

● Transformed expression values

○ Normalized expression values

Permutations for p-value calculation

Number: 100

Figure 20.53: *Gene set enrichment analsysis parameters.*

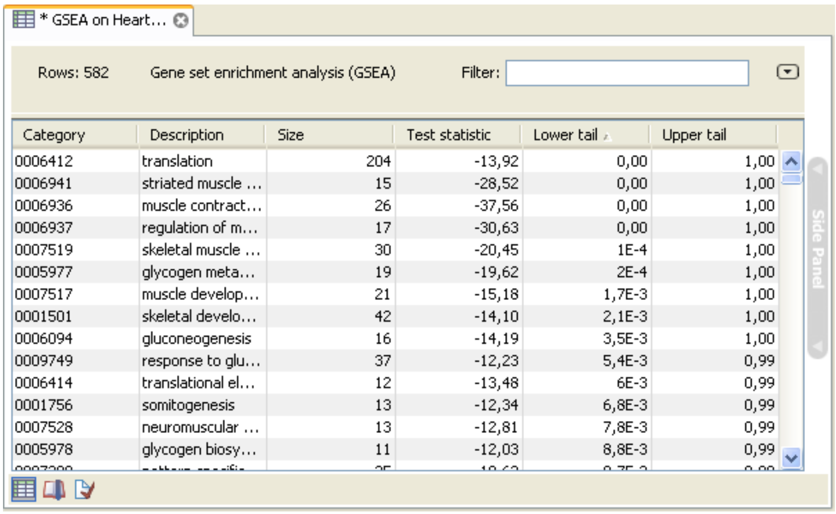
At the top, you can select which values to analyze (see section 20.2.1).

Below, you can set the **Permutations for p-value calculation**. For the GSEA test a p-value is calculated by permutation: p permuted data sets are generated, each consisting of the original features, but with the test statistics permuted. The GSEA test is run on each of the permuted data sets. The test statistic is calculated on the original data, and the resulting value is compared to the distribution of the values obtained for the permuted data sets. The permutation based p-value is the number of permutation based test statistics above (or below) the value of the test statistic for the original data, divided by the number of permuted data sets. For reliable permutation-based p-value calculation a large number of permutations is required (100 is the default).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Result of gene set enrichment analysis

The result of performing gene set enrichment analysis using GO biological process is shown in figure 20.54.



Rows: 582 Gene set enrichment analysis (GSEA) Filter:

Category	Description	Size	Test statistic	Lower tail	Upper tail
0006412	translation	204	-13,92	0,00	1,00
0006941	striated muscle ...	15	-28,52	0,00	1,00
0006936	muscle contract...	26	-37,56	0,00	1,00
0006937	regulation of m...	17	-30,63	0,00	1,00
0007519	skeletal muscle ...	30	-20,45	1E-4	1,00
0005977	glycogen meta...	19	-19,62	2E-4	1,00
0007517	muscle develop...	21	-15,18	1,7E-3	1,00
0001501	skeletal develo...	42	-14,10	2,1E-3	1,00
0006094	gluconeogenesis	16	-14,19	3,5E-3	1,00
0009749	response to glu...	37	-12,23	5,4E-3	0,99
0006414	translational el...	12	-13,48	6E-3	0,99
0001756	somitogenesis	13	-12,34	6,8E-3	0,99
0007528	neuromuscular ...	13	-12,81	7,8E-3	0,99
0005978	glycogen biosy...	11	-12,03	8,8E-3	0,99

Figure 20.54: *The result of gene set enrichment analysis on GO biological process.*

The table shows the following information:

- **Category**. This is the identifier for the category.
- **Description**. This is the description belonging to the category. Both of these are simply extracted from the annotations.
- **Size**. The number of features with this category. (Note that this is after removal of duplicates).
- **Test statistic**. This is the GSEA test statistic.
- **Lower tail**. This is the mass in the permutation based p-value distribution below the value of the test statistic.
- **Upper tail**. This is the mass in the permutation based p-value distribution above the value of the test statistic.

A small lower (or upper) tail p-value for an annotation category is an indication that features in this category viewed as a whole are perturbed among the groups in the experiment considered.

20.7 General plots

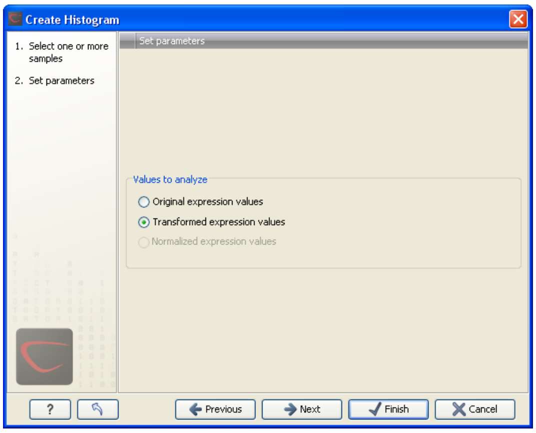
The last folder in the **Expression Analysis** () folder in the **Toolbox** is **General Plots**. Here you find three general plots that may be useful at various point of your analysis work flow. The plots are explained in detail below.

20.7.1 Histogram

A histogram shows a distribution of a set of values. Histograms are often used for examining and comparing distributions, e.g. of expression values of different samples, in the quality control step of an analysis. You can create a histogram showing the distribution of expression value for a sample:

Toolbox | **Expression Analysis (**)** | **General Plots** | **Create Histogram (**)**

Select a number of samples (() or ()). When you have selected more than one sample, a histogram will be created for each one. Clicking **Next** will display a dialog as shown in figure 20.55.



Values to analyze
○ Original expression values
● Transformed expression values
○ Normalized expression values

Figure 20.55: *Selcting which values the histogram should be based on.*

In this dialog, you select the values to be used for creating the histogram (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Viewing histograms

The resulting histogram is shown in a figure 20.56

The histogram shows the expression value on the x axis (in the case of figure 20.56 the transformed expression values) and the counts of these values on the y axis.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.

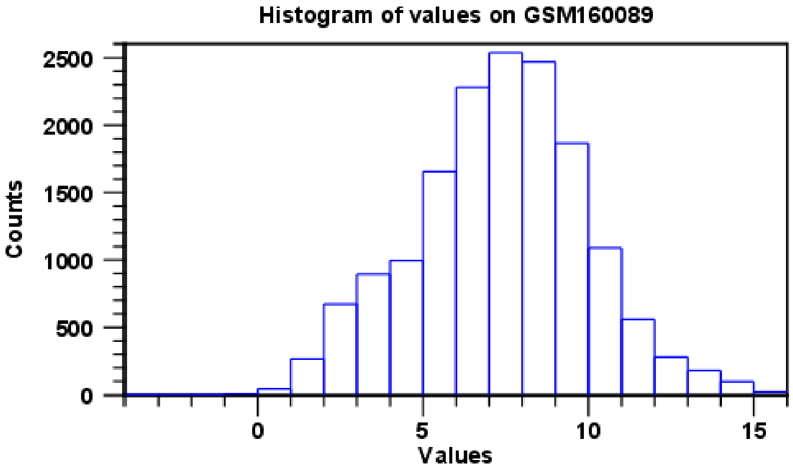


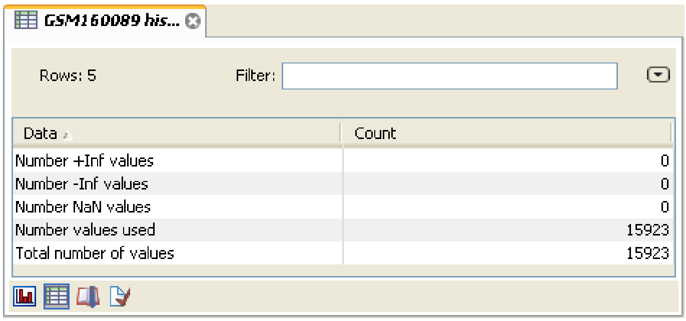
Figure 20.56: *Histogram showing the distribution of transformed expression values.*

- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Break points**. Determines where the bars in the histogram should be:
  - **Sturges method**. This is the default. The number of bars is calculated from the range of values by Sturges formula [Sturges, 1926].
  - **Equi-distanced bars**. This will show bars from **Start** to **End** and with a width of **Sep**.
  - **Number of bars**. This will simply create a number of bars starting at the lowest value and ending at the highest value.

Below the graph preferences, you find **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a principal component plot, you need to save the settings of the **Side Panel** (see section 5.6).

Besides the histogram view itself, the histogram can also be shown in a table, summarizing key properties of the expression values. An example is shown in figure 20.57.



Data	Count
Number +Inf values	0
Number -Inf values	0
Number NaN values	0
Number values used	15923
Total number of values	15923

Figure 20.57: *Table view of a histogram.*

The table lists the following properties:

- **Number +Inf values**
- **Number -Inf values**
- **Number NaN values**
- **Number values used**
- **Total number of values**

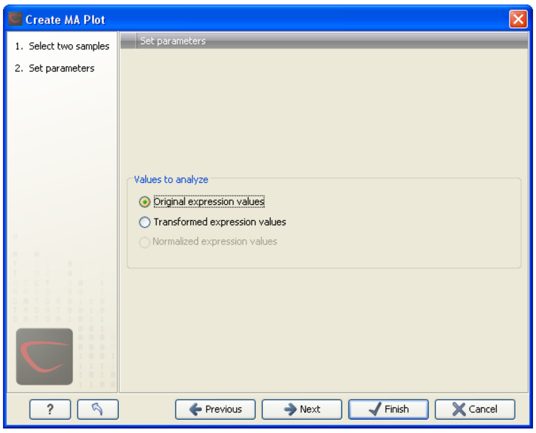
20.7.2 MA plot

The MA plot is a scatter rotated by $45°$. For two samples of expression values it plots for each gene the difference in expression against the mean expression level. MA plots are often used for quality control, in particular, to assess whether normalization and/or transformation is required.

You can create an MA plot comparing two samples:

Toolbox | **Expression Analysis ()** | **General Plots** | **Create MA Plot ()**

Select two samples (() or ()). Clicking **Next** will display a dialog as shown in figure 20.58.



Values to analyze

● Original expression values
○ Transformed expression values
○ Normalized expression values

Figure 20.58: *Selcting which values the MA plot should be based on.*

In this dialog, you select the values to be used for creating the MA plot (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

Viewing MA plots

The resulting plot is shown in a figure 20.59.

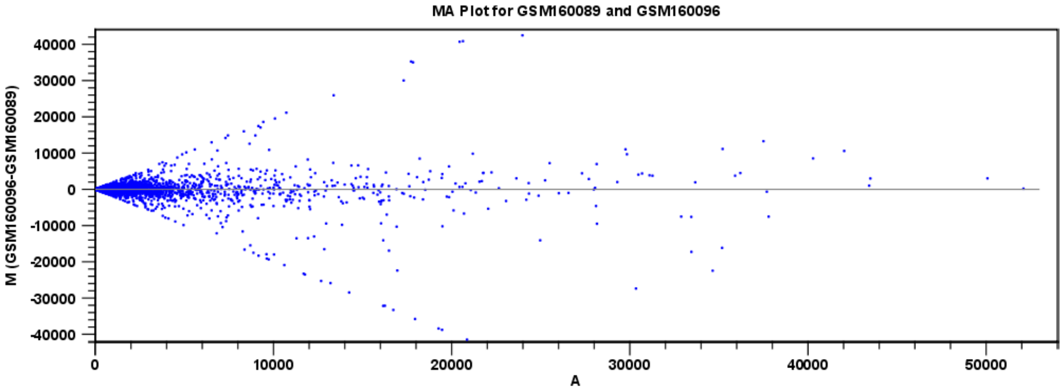


Figure 20.59: *MA plot based on original expression values.*

The X axis shows the mean expression level of a feature on the two samples and the Y axis shows the difference in expression levels for a feature on the two samples. From the plot shown in figure 20.59 it is clear that the variance increases with the mean. With an MA plot like this, you will often choose to transform the expression values (see section 20.2.2).

Figure 20.60 shows the same two samples where the MA plot has been created using log2 transformed values.

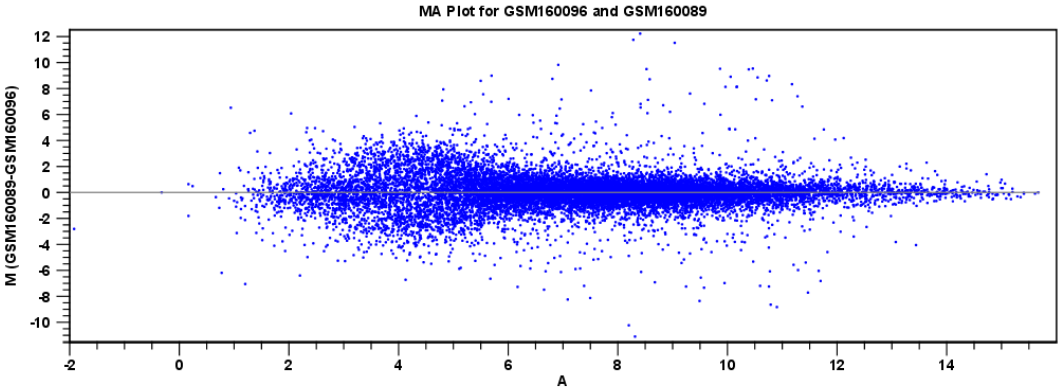


Figure 20.60: *MA plot based on transformed expression values.*

The much more symmetric and even spread indicates that the dependance of the variance on the mean is not as strong as it was before transformation.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.

- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **y = 0 axis**. Draws a line where y = 0. Below there are some options to control the appearance of the line:
  - **Line width**
    - Thin
    - Medium
    - Wide
  - **Line type**
    - None
    - Line
    - Long dash
    - Short dash
  - **Line color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Line width**
  - Thin
  - Medium
  - Wide
- **Line type**
  - None
  - Line
  - Long dash
  - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties** preferences, where you can adjust coloring and appearance of the dots:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

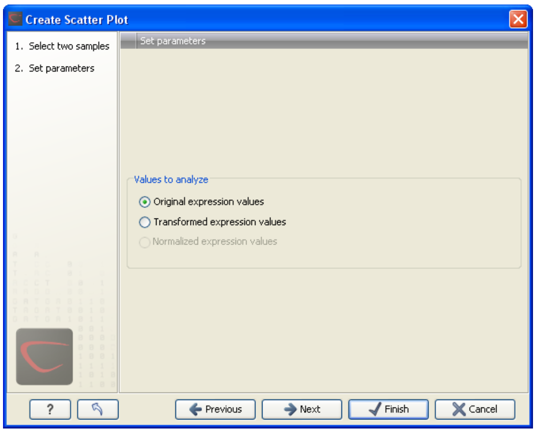
Note that if you wish to use the same settings next time you open a scatter plot, you need to save the settings of the **Side Panel** (see section 5.6).

20.7.3 Scatter plot

As described in section 20.1.5, an experiment can be viewed as a scatter plot. However, you can also create a "stand-alone" scatter plot of two samples:

Toolbox | **Expression Analysis ()** | **General Plots** | **Create Scatter Plot ()**

Select two samples (() or ()). Clicking **Next** will display a dialog as shown in figure 20.61.



Values to analyze

● Original expression values

○ Transformed expression values

○ Normalized expression values

Figure 20.61: *Selcting which values the scatter plot should be based on.*

In this dialog, you select the values to be used for creating the scatter plot (see section 20.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

For more information about the scatter plot view and how to interpret it, please see section 20.1.5.

Chapter 21

Cloning and cutting

Contents

CLC Genomics Workbench offers graphically advanced *in silico* cloning and design of vectors for various purposes together with restriction enzyme analysis and functionalities for managing lists of restriction enzymes.

First, after a brief introduction, restriction cloning and general vector design is explained. Next, we describe how to do Gateway Cloning [1]. Finally, the general restriction site analyses are described.

[1] Gateway is a registered trademark of Invitrogen Corporation

21.1 Molecular cloning

Molecular cloning is a very important tool in the quest to understand gene function and regulation. Through molecular cloning it is possible to study individual genes in a controlled environment. Using molecular cloning it is possible to build complete libraries of fragments of DNA inserted into appropriate cloning vectors.

The *in silico* cloning process in *CLC Genomics Workbench* begins with the selection of sequences to be used:

Toolbox | **Cloning and Restriction Sites** ( ) | **Cloning** ( )

This will open a dialog where you can select the sequences containing the fragments you want to clone (figure 21.1).

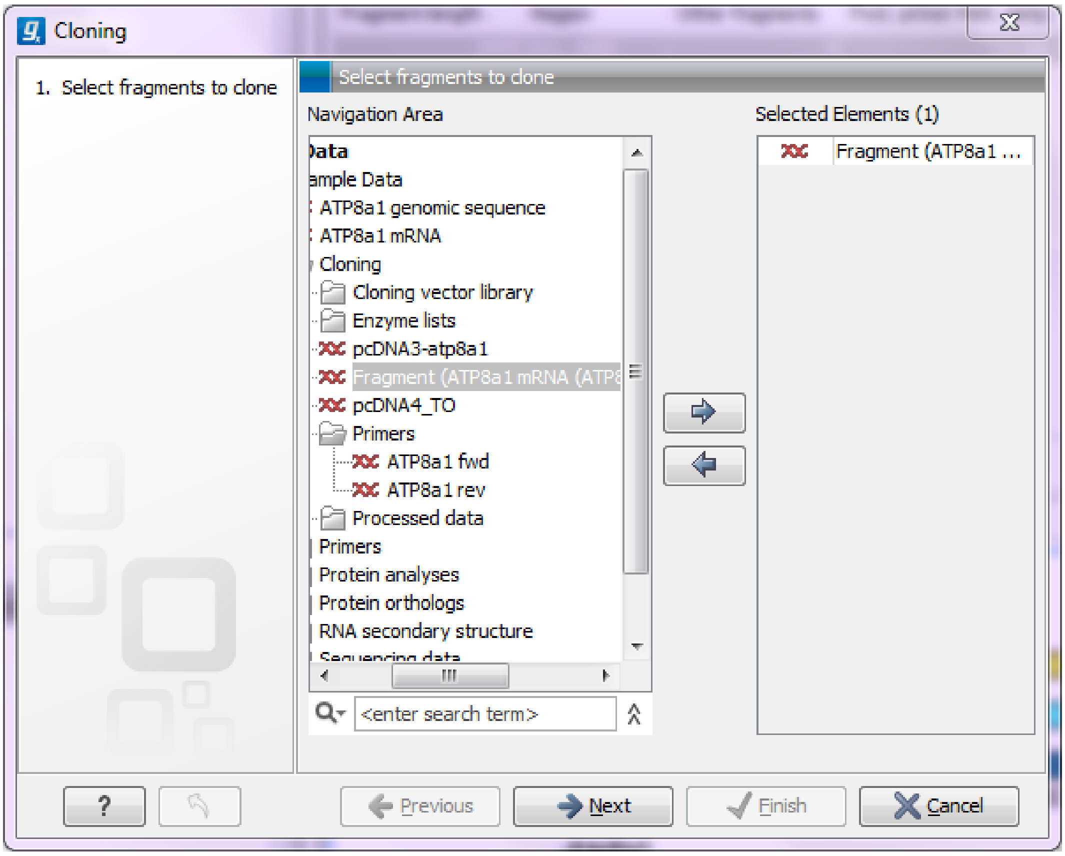


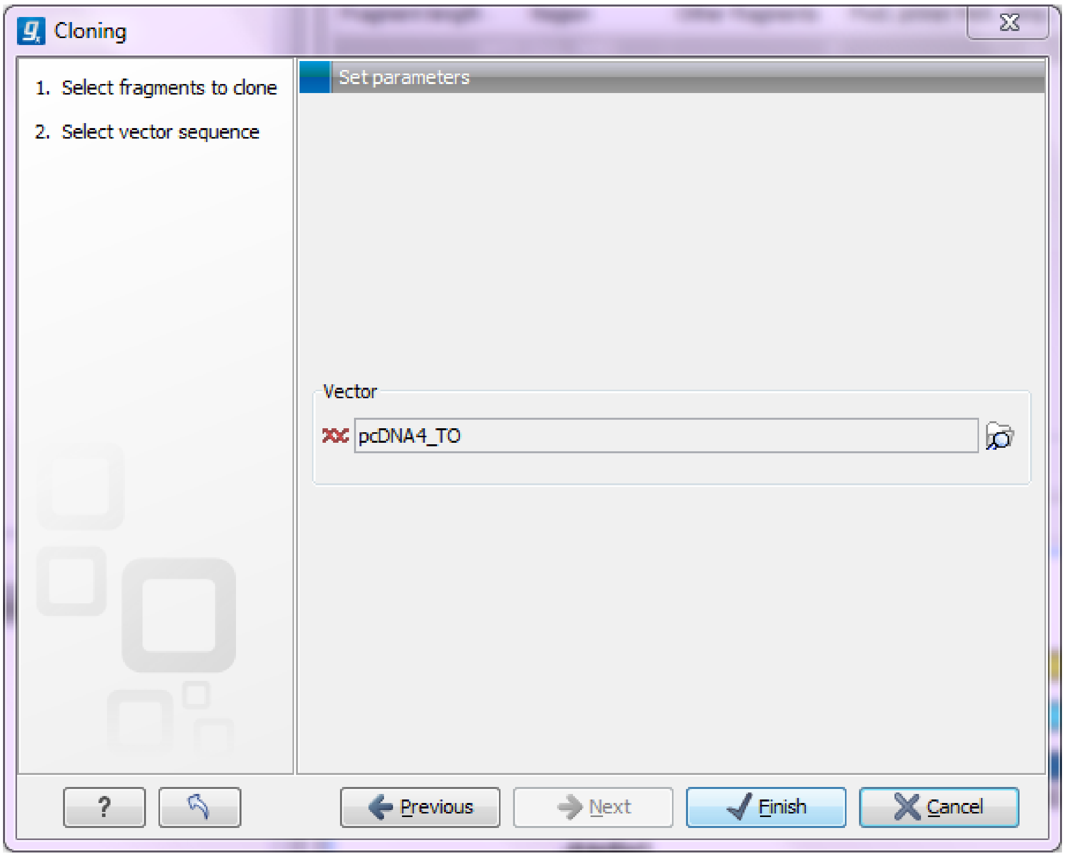
Figure 21.1: *Selecting one or more sequences containing the fragments you want to clone.*

Note that the vector sequence will be selected when you click **Next** as shown in figure figure 21.2.

Select the cloning vector by clicking the browse ( ) button. Once the sequence has been selected, click **Finish**. The *CLC Genomics Workbench* will now create a sequence list of the fragments and vector sequences and open it in the cloning editor as shown in figure 21.3.

When you save the cloning experiment, it is saved as a **Sequence list**. See section 10.7 for more information about sequence lists. If you need to open the list later for cloning work, simply switch to the **Cloning** ( ) editor at the bottom of the view.

If you later in the process need additional sequences, you can easily add more sequences to the



Vector

pcDNA4_TO

Figure 21.2: *Selecting a cloning vector.*

view. Just:

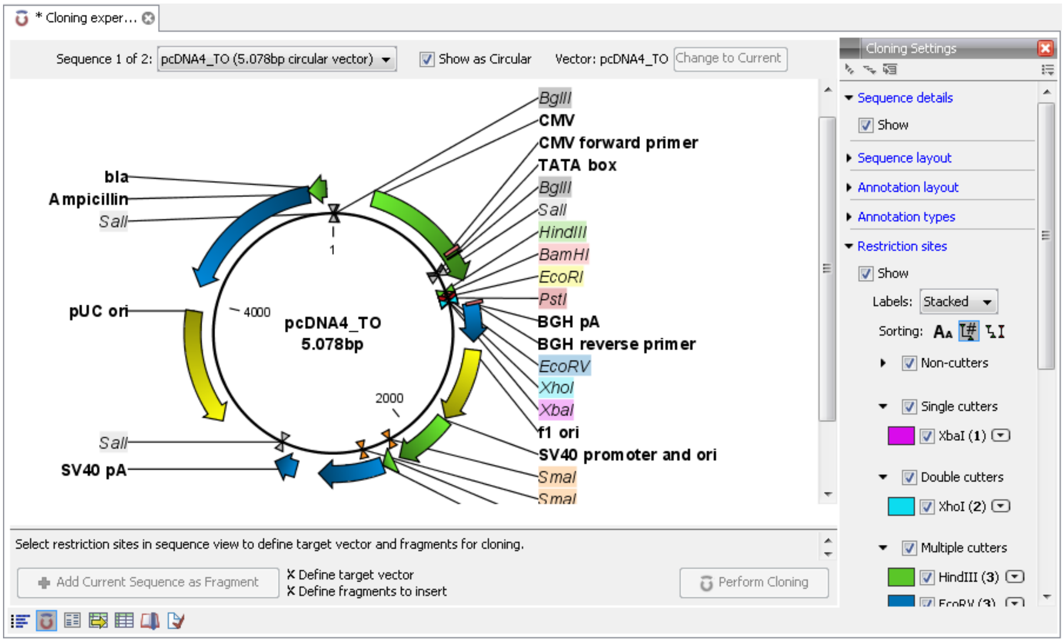
right-click anywhere on the empty white area | Add Sequences

21.1.1 Introduction to the cloning editor

In the cloning editor, most of the basic options for viewing, selecting and zooming the sequences are the same as for the standard sequence view. See section 10.1 for an explanation of these options. This means that e.g. known SNP's, exons and other annotations can be displayed on the sequences to guide the choice of regions to clone.

However, the cloning editor has a special layout with three distinct areas (in addition to the **Side Panel** found in other sequence views as well):

- At the top, there is a panel to switch between the sequences selected as input for the cloning. You can also specify whether the sequence should be visualized **as circular** or as a fragment. At the right-hand side, there is a button to the status of the sequence currently shown to **vector**.
- In the middle, the selected sequence is shown. This is the central area for defining how the cloning should be performed. This is explained in details below.
- At the bottom, there is a panel where the selection of fragments and target vector is performed (see elaboration below).



Sequence 1 of 2: pcDNA4_TO (5.078bp circular vector)

☑ Show as Circular

Vector: pcDNA4_TO

Select restriction sites in sequence view to define target vector and fragments for cloning.

✕ Define target vector
✕ Define fragments to insert

Labels: Stacked

Figure 21.3: *Cloning editor.*

There are essentially three ways of performing cloning in the *CLC Genomics Workbench*. The *first* is the most straight-forward approach which is based on a simple model of selecting restriction sites for cutting out one or more fragments and defining how to open the vector to insert the fragments. This is described as *the cloning work flow* below. The *second* approach is unguided and more flexible and allows you to manually cut, copy, insert and replace parts of the sequences. This approach is described under *manual cloning* below. *Finally*, the *CLC Genomics Workbench* also supports *Gateway cloning* (see section 21.2).

21.1.2 The cloning work flow

The *cloning work flow* is designed to support restriction cloning work flows through the following steps:

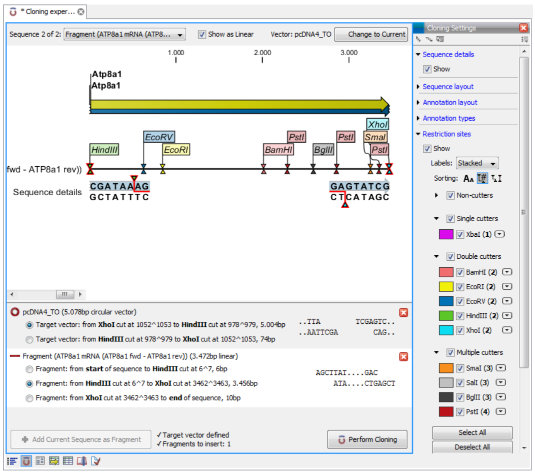
1. Define one or more fragments
2. Define how the vector should be opened
3. Specify orientation and order of the fragment

Defining fragments

First, select the sequence containing the cloning fragment in the list at the top of the view. Next, make sure the restriction enzyme you wish to use is listed in the **Side Panel** (see section 21.3.1). To specify which part of the sequence should be treated as the fragment, first click one of the cut sites you wish to use. Then press and hold the Ctrl key (⌘ on Mac) while you click the second cut site. You can also right-click the cut sites and use the **Select This ... Site** to select a site.

When this is done, the panel below will update to reflect the selections (see figure 21.4).

In this example you can see that there are now three options listed in the panel below the view.



pcDNA4_TO (5.078bp circular vector)

● Target vector: from **XhoI** cut at 1052^1053 to **HindIII** cut at 978^979, 5.004bp

○ Target vector: from **HindIII** cut at 978^979 to **XhoI** cut at 1052^1053, 74bp

```
..TTA          TCGAGTC..
..AATTCGA          CAG..
```

Fragment (ATP8a1 mRNA (ATP8a1 fwd - ATP8a1 rev)) (3.472bp linear)

○ Fragment: from **start** of sequence to **HindIII** cut at 6^7, 6bp

● Fragment: from **HindIII** cut at 6^7 to **XhoI** cut at 3462^3463, 3.456bp

○ Fragment: from **XhoI** cut at 3462^3463 to **end** of sequence, 10bp

```
AGCTTAT....GAC
   ATA....CTGAGCT
```

✓Target vector defined
✓Fragments to insert: 1

Figure 21.4: *HindIII and XhoI cut sites selected to cut out fragment.*

This is because there are now three options for selecting the fragment that should be used for cloning. The fragment selected per default is the one that is in between the cut sites selected.

If the entire sequence should be selected as fragment, click the **Add Current Sequence as Fragment** (✚).

At any time, the selection of cut sites can be cleared by clicking the **Remove** (☒) icon to the right of the fragment selections. If you just wish to remove the selection of one of the sites, right-click the site on the sequence and choose **De-select This ... Site**.

Defining target vector

When selecting among the sequences in the panel at the top, the vector sequence has "vector" appended to its name. If you wish to use one of the other sequences as vector, select this sequence in the list and click **Change to Current**.

The next step is to define where the vector should be cut. If the vector sequence should just be opened, click the restriction site you want to use for opening. If you want to cut off part of the vector, click two restriction sites while pressing the Ctrl key (⌘ on Mac). You can also right-click the cut sites and use the **Select This ... Site** to select a site.

This will display two options for what the target vector should be (for linear vectors there would have been three option) as shown in figure 21.5)

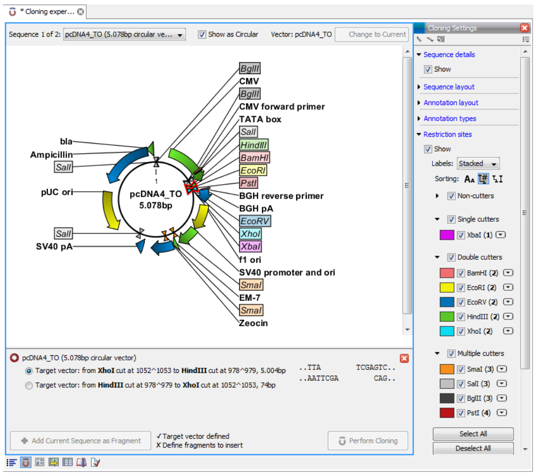
Just as when cutting out the fragment, there is a lost of choices regarding which sequence should be used as the vector.

At any time, the selection of cut sites can be cleared by clicking the **Remove** (☒) icon to the right of the target vector selections. If you just wish to remove the selection of one of the sites, right-click the site on the sequence and choose **De-select This ... Site**.

When the right target vector is selected, you are ready to **Perform Cloning** (ʊ), see below.

Perform cloning

Once selections have been made for both fragments and vector, click **Perform Cloning** (ʊ). This will display a dialog to adapt overhangs and change orientation as shown in figure 21.6)



pcDNA4_TO (5.078bp circular vector)

● Target vector: from **XhoI** cut at 1052^1053 to **HindIII** cut at 978^979, 5.004bp

○ Target vector: from **HindIII** cut at 978^979 to **XhoI** cut at 1052^1053, 74bp

```
..TTA          TCGAGTC..
..AATTCGA          CAG..
```

✓ Target vector defined
X Define fragments to insert

Figure 21.5: *HindIII and XhoI sites used to open the vector.*

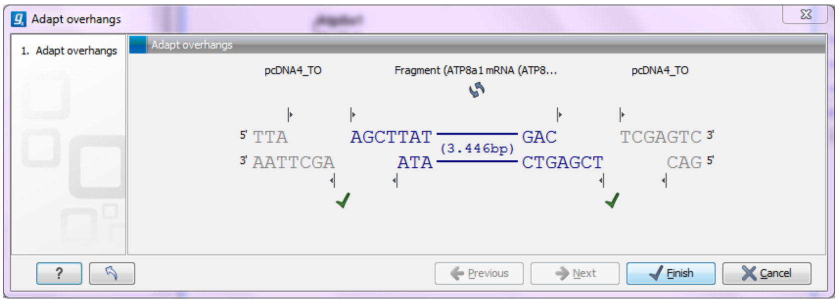


Figure 21.6: *Showing the insertion point of the vector.*

This dialog visualizes the details of the insertion. The vector sequence is on each side shown in a faded gray color. In the middle the fragment is displayed. If the overhangs of the sequence and the vector do not match, you can blunt end or fill in the overhangs using the **drag handles** (⫞). Click and drag with the mouse to adjust the overhangs.

Whenever you drag the handles, the status of the insertion point is indicated below:

- The overhangs match (✓).
- The overhangs do not match (🚫). In this case, you will not be able to click **Finish**. Drag the handles to make the overhangs match.

The fragment can be reverse complemented by clicking the **Reverse complement fragment** (⟲).

When several fragments are used, the order of the fragments can be changed by clicking the move buttons (⇨)/ (⇦).

When you click **Finish** the final construct will be shown (see figure 21.7).

You can now **Save** (💾) this sequence for later use. The cloning experiment used to design the construct can be saved as well. If you check the **History** (📖) of the construct, you can see the details about restriction sites and fragments used for the cloning.

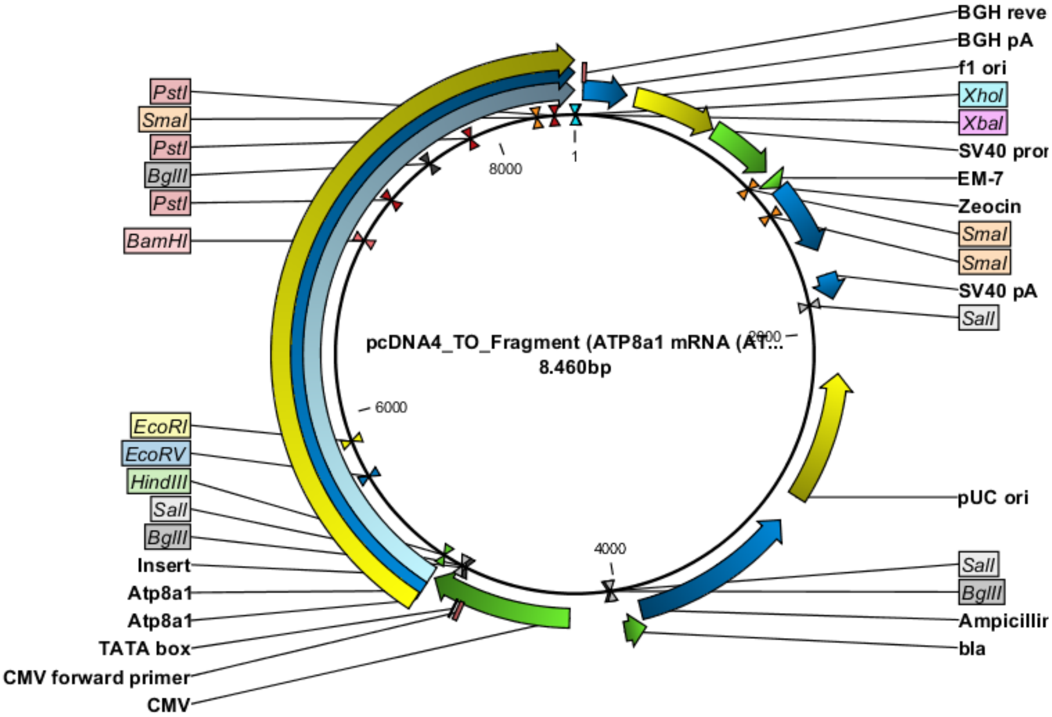


Figure 21.7: *The final construct.*

21.1.3 Manual cloning

If you wish to use the manual way of cloning (as opposed to using the cloning work flow explained above in section 21.1.2), you can disregard the panel at the bottom. The manual cloning approach is based on a number of ways that you can manipulate the sequences. All manipulations of sequences are done manually, giving you full control over how the final construct is made. Manipulations are performed through right-click menus which have three different appearances depending on where you click, as visualized in figure 21.8.

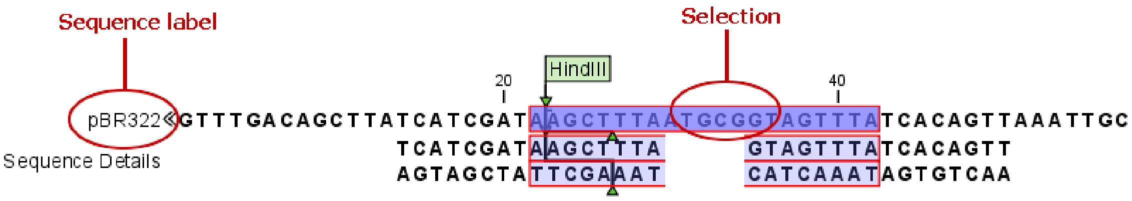


Figure 21.8: *The red circles mark the two places you can use for manipulating the sequences.*

- **Right-click the sequence name (to the left) to manipulate the whole sequence.**
- **Right-click a selection to manipulate the selection.**

The two menus are described in the following:

Manipulate the whole sequence

Right-clicking the sequence name at the left side of the view reveals several options on sorting, opening and editing the sequences in the view (see figure 21.9).

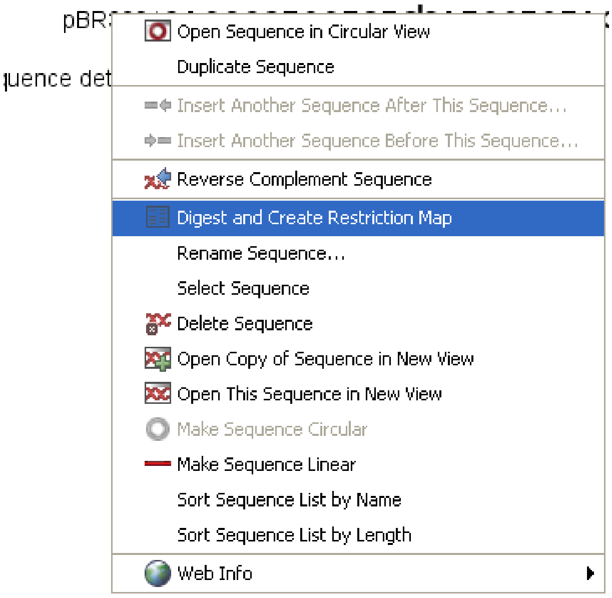

pBR
quence det
Open Sequence in Circular View
Duplicate Sequence
Insert Another Sequence After This Sequence...
Insert Another Sequence Before This Sequence...
Reverse Complement Sequence
Digest and Create Restriction Map
Rename Sequence...
Select Sequence
Delete Sequence
Open Copy of Sequence in New View
Open This Sequence in New View
Make Sequence Circular
Make Sequence Linear
Sort Sequence List by Name
Sort Sequence List by Length
Web Info

Figure 21.9: *Right click on the sequence in the cloning view.*

- **Open sequence in circular view** (○)
  Opens the sequence in a new circular view. If the sequence is not circular, you will be asked if you wish to make it circular or not. (This will not forge ends with matching overhangs together - use "Make Sequence Circular" (○) instead.)

- **Duplicate sequence**
  Adds a duplicate of the selected sequence. The new sequence will be added to the list of sequences shown on the screen.

- **Insert sequence after this sequence** (▬◆)
  Insert another sequence after this sequence. The sequence to be inserted can be selected from a list which contains the sequences present in the cloning editor. The inserted sequence remains on the list of sequences. If the two sequences do not have blunt ends, the ends' overhangs have to match each other. Otherwise a warning is displayed.

- **Insert sequence before this sequence** (◆▬)
  Insert another sequence before this sequence. The sequence to be inserted can be selected from a list which contains the sequences present in the cloning editor. The inserted sequence remains on the list of sequences. If the two sequences do not have blunt ends, the ends' overhangs have to match each other. Otherwise a warning is displayed.

- **Reverse sequence**
  Reverse the sequence and replaces the original sequence in the list. This is sometimes useful when working with single stranded sequences. Note that this is *not* the same as creating the reverse *complement* (see the following item in the list).

- **Reverse complement sequence** (✗)
  Creates the reverse complement of a sequence and replaces the original sequence in the list. This is useful if the vector and the insert sequences are not oriented the same way.

- **Digest Sequence with Selected Enzymes and Run on Gel** (☰)
  See section 21.4.1

- **Rename sequence**
  Renames the sequence.
- **Select sequence**
  This will select the entire sequence.
- **Delete sequence ()**
  This deletes the given sequence from the cloning editor.
- **Open copy of sequencew ()**
  This will open a copy of the selected sequence in a normal sequence view.
- **Open this sequence ()**
  This will open the selected sequence in a normal sequence view.
- **Make sequence circular** ()
  This will convert a sequence from a linear to a circular form. If the sequence have matching overhangs at the ends, they will be merged together. If the sequence have incompatible overhangs, a dialog is displayed, and the sequence cannot be made circular. The circular form is represented by >> and << at the ends of the sequence.
- **Make sequence linear ()**
  This will convert a sequence from a circular to a linear form, removing the << and >> at the ends.

Manipulate parts of the sequence

Right-clicking a selection reveals several options on manipulating the selection (see figure 21.10).

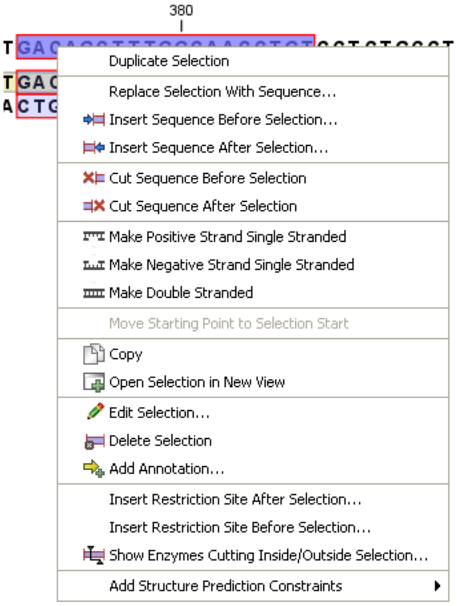

380
Duplicate Selection
Replace Selection With Sequence...
Insert Sequence Before Selection...
Insert Sequence After Selection...
Cut Sequence Before Selection
Cut Sequence After Selection
Make Positive Strand Single Stranded
Make Negative Strand Single Stranded
Make Double Stranded
Move Starting Point to Selection Start
Copy
Open Selection in New View
Edit Selection...
Delete Selection
Add Annotation...
Insert Restriction Site After Selection...
Insert Restriction Site Before Selection...
Show Enzymes Cutting Inside/Outside Selection...
Add Structure Prediction Constraints

Figure 21.10: *Right click on a sequence selection in the cloning view.*

- **Replace Selection with sequence.** This will replace the selected region with a sequence. The sequence to be inserted can be selected from a list containing all sequences in the cloning editor.

- **Insert Sequence before Selection ().** Insert a sequence before the selected region. The sequence to be inserted can be selected from a list containing all sequences in the cloning editor.

- **Insert Sequence after Selection ().** Insert a sequence after the selected region. The sequence to be inserted can be selected from a list containing all sequences in the cloning editor.

- **Cut Sequence before Selection ().** This will cleave the sequence before the selection and will result in two smaller fragments.

- **Cut Sequence after Selection ().** This will cleave the sequence after the selection and will result in two smaller fragments.

- **Make Positive Strand Single Stranded ().** This will make the positive strand of the selected region single stranded.

- **Make Negative Strand Single Stranded ().** This will make the negative strand of the selected region single stranded.

- **Make Double Stranded ().** This will make the selected region double stranded.

- **Move Starting Point to Selection Start.** This is only active for circular sequences. It will move the starting point of the sequence to the beginning of the selection.

- **Copy Selection ().** This will copy the selected region to the clipboard, which will enable it for use in other programs.

- **Duplicate Selection.** If a selection on the sequence is duplicated, the selected region will be added as a new sequence to the cloning editor with a new sequence name representing the length of the fragment. When a sequence region between two restriction sites are double-clicked the entire region will automatically be selected. This makes it very easy to make a new sequence from a fragment created by cutting with two restriction sites (right-click the selection and choose **Duplicate selection**).

- **Open Selection in New View ().** This will open the selected region in the normal sequence view.

- **Edit Selection ().** This will open a dialog box, in which is it possible to edit the selected residues.

- **Delete Selection ().** This will delete the selected region of the sequence.

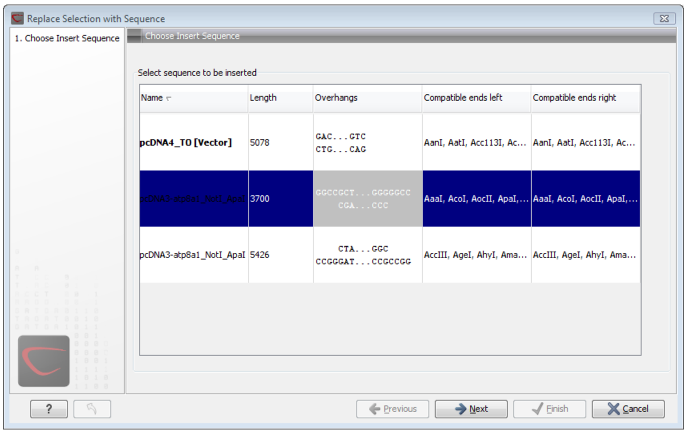
- **Add Annotation ().** This will open the **Add annotation** dialog box.

- **Show Enzymes Only Cutting Selection ().** This will add enzymes cutting this selection to the Side Panel.

- **Insert Restriction Sites before/after Selection.** This will show a dialog where you can choose from a list restriction enzymes (see section 21.1.4).

Insert one sequence into another

Sequences can be inserted into each other in several ways as described in the lists above. When you chose to insert one sequence into another you will be presented with a dialog where all sequences in the view are present (see figure 21.11).


Replace Selection with Sequence
1. Choose Insert Sequence
Choose Insert Sequence
? | Previous | Next | Finish | Cancel

Select sequence to be inserted

Name	Length	Overhangs	Compatible ends left	Compatible ends right
pcDNA4_TO [Vector]	5078	GAC...GTC CTG...CAG	AanI, AatI, Acc113I, Ac...	AanI, AatI, Acc113I, Ac...
pcDNA3-atp8a1_NotI_ApaI	3700	GGCCGCT...GGGGGCC CGA...CCC	AaaI, AcoI, AocII, ApaI,...	AaaI, AcoI, AocII, ApaI,...
pcDNA3-atp8a1_NotI_ApaI	5426	CTA...GGC CCGGGAT...CCGCCGG	AccIII, AgeI, AhyI, Ama...	AccIII, AgeI, AhyI, Ama...

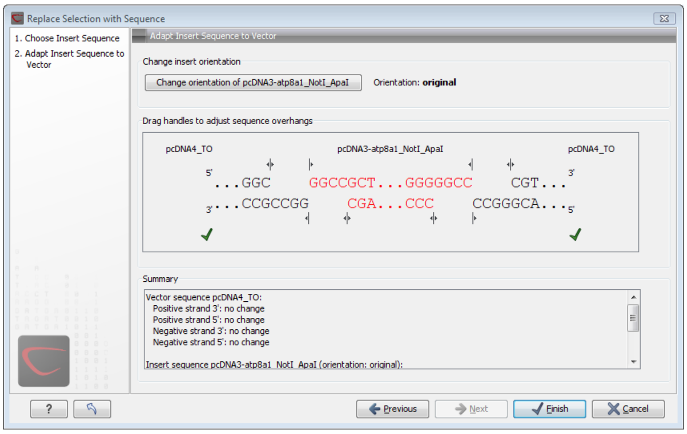
Figure 21.11: *Select a sequence for insertion.*

The sequence that you have chosen to insert into will be marked with **bold** and the text **[vector]** is appended to the sequence name. Note that this is completely unrelated to the vector concept in the cloning work flow described in section 21.1.2.

The list furthermore includes the length of the fragment, an indication of the overhangs, and a list of enzymes that are compatible with this overhang (for the left and right ends, respectively). If not all the enzymes can be shown, place your mouse cursor on the enzymes, and a full list will be shown in the tool tip.

Select the sequence you wish to insert and click **Next**.

This will show the dialog in figure 21.12).


Replace Selection with Sequence
1. Choose Insert Sequence
2. Adapt Insert Sequence to Vector
Adapt Insert Sequence to Vector
Change insert orientation
Change orientation of pcDNA3-atp8a1_NotI_ApaI
Drag handles to adjust sequence overhangs
pcDNA4_TO
pcDNA3-atp8a1_NotI_ApaI
pcDNA4_TO
5'
3'
...GGC
GGCCGCT...GGGGGCC
CGT...
3'
5'
...CCGCCGG
CGA...CCC
CCGGGCA...
Summary
? | Previous | Next | Finish | Cancel

Orientation: **original**

Vector sequence pcDNA4_TO:
Positive strand 3': no change
Positive strand 5': no change
Negative strand 3': no change
Negative strand 5': no change

Insert sequence pcDNA3-atp8a1_NotI_ApaI (orientation: original):

Figure 21.12: *Drag the handles to adjust overhangs.*

At the top is a button to reverse complement the inserted sequence.

Below is a visualization of the insertion details. The inserted sequence is at the middle shown in red, and the vector has been split at the insertion point and the ends are shown at each side of the inserted sequence.

If the overhangs of the sequence and the vector do not match, you can blunt end or fill in the overhangs using the **drag handles** (⁞).

Whenever you drag the handles, the status of the insertion point is indicated below:

- The overhangs match (✓).
- The overhangs do not match (⊘). In this case, you will not be able to click **Finish**. Drag the handles to make the overhangs match.

At the bottom of the dialog is a summary field which records all the changes made to the overhangs. This contents of the summary will also be written in the history (📖) when you click **Finish**.

When you click **Finish** and the sequence is inserted, it will be marked with a selection.

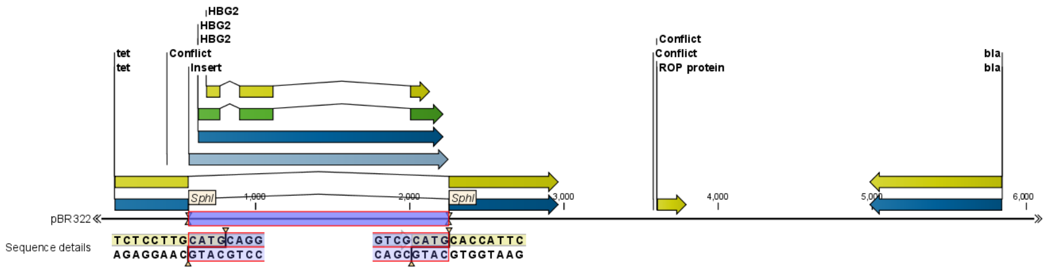


Figure 21.13: *One sequence is now inserted into the cloning vector. The sequence inserted is automatically selected.*

21.1.4 Insert restriction site

If you make a selection on the sequence, right-click, you find this option for inserting the recognition sequence of a restriction enzyme before or after the region you selected. This will display a dialog as shown in figure 21.14

At the top, you can select an existing enzyme list or you can use the full list of enzymes (default). Select an enzyme, and you will see its recognition sequence in the text field below the list (AAGCTT). If you wish to insert additional residues such as tags etc., this can be typed into the text fields adjacent to the recognition sequence. .

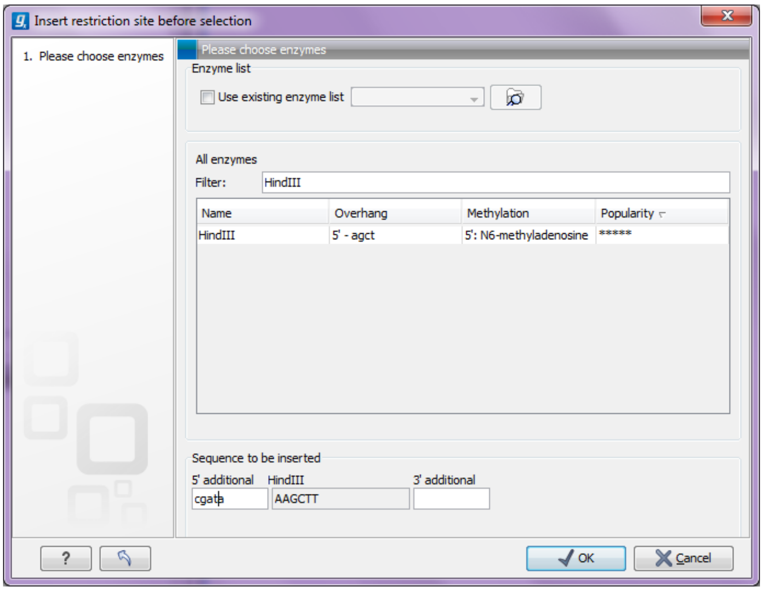
Click **OK** will insert the sequence before or after the selection. If the enzyme selected was not already present in the list in the **Side Panel**, it will now be added and selected. Furthermore, an restriction site annotation is added.

21.2 Gateway cloning

CLC Genomics Workbench offers tools to perform *in silico* Gateway cloning[2]. Multi-site Gateway cloning is not supported.

The three tools for doing Gateway cloning in the *CLC Genomics Workbench* mimic the procedure followed in the lab:

[2] Gateway is a registered trademark of Invitrogen Corporation



Enzyme list

□ Use existing enzyme list

All enzymes

Filter: HindIII

Name	Overhang	Methylation	Popularity
HindIII	5' - agct	5': N6-methyladenosine	*****

Sequence to be inserted

5' additional	HindIII	3' additional
cgata	AAGCTT	

Figure 21.14: *Inserting the HindIII recognition sequence.*

- First, attB sites are added to a sequence fragment
- Second, the attB-flanked fragment is recombined into a donor vector (the BP reaction) to construct an entry clone
- Finally, the target fragment from the entry clone is recombined into an expression vector (the LR reaction) to construct an expression clone

During this process, both the attB-flanked fragment and the entry clone can be saved.

For more information about the Gateway technology, please visit http://www.invitrogen.com/site/us/en/home/Products-and-Services/Applications/Cloning/Gateway-Cloning.html

To perform these analyses in the *CLC Genomics Workbench*, you need to import donor and expression vectors. These can be downloaded from Invitrogen's web site and directly imported into the Workbench: http://tools.invitrogen.com/downloads/Gateway%20vectors.ma4

21.2.1 Add attB sites

The first step in the Gateway cloning process is to amplify the target sequence with primers including so-called attB sites. In the *CLC Genomics Workbench*, you can add attB sites to a sequence fragment in this way:

> **Toolbox in the Menu Bar** | **Cloning and Restriction Sites** ( ) | **Gateway Cloning** ( ) | **Add attB Sites** ( )

This will open a dialog where you can select on ore more sequences. Note that if your fragment is part of a longer sequence, you need to extract it first. This can be done in two ways:

- If the fragment is covered by an annotation (if you want to use e.g. a CDS), simply right-click the annotation and **Open Annotation in New View**
- Otherwise you can simply make a selection on the sequence, right-click and **Open Selection in New View**

In both cases, the selected part of the sequence will be copied and opened as a new sequence which can be **Saved** (🖫).

When you have selected your fragment(s), click **Next**.

This will give you options to extend the fragment with additional sequences by extending the primers 5' of the template-specific part of the primer (i.e. between the template specific part and the attB sites). See an example of this in figure 21.20 where a Shine-Dalgarno site has been added between the attB site and the gene of interest.

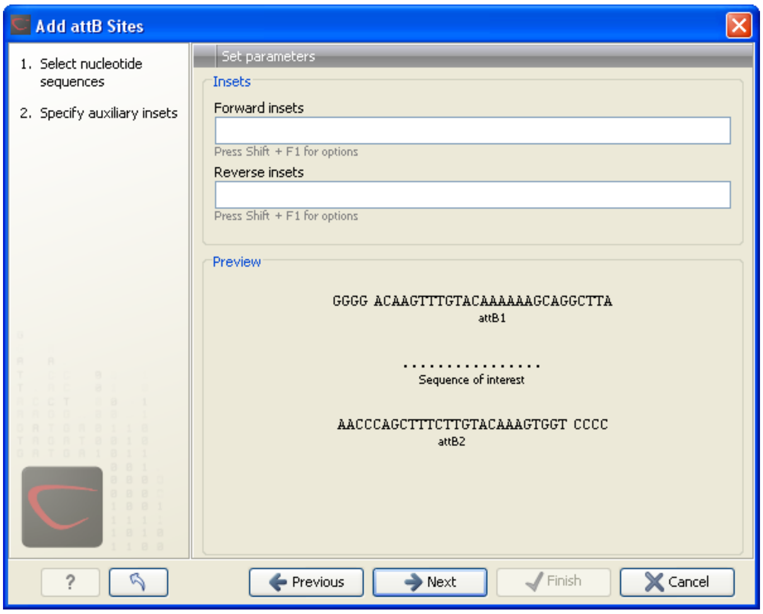

Add attB Sites
1. Select nucleotide sequences
2. Specify auxiliary insets
Set parameters
Insets
Forward insets
Press Shift + F1 for options
Reverse insets
Press Shift + F1 for options
Preview
GGGG ACAAGTTTGTACAAAAAAGCAGGCTTA
attB1
....
Sequence of interest
AACCCAGCTTTCTTGTACAAAGTGGT CCCC
attB2
?
Previous
Next
Finish
Cancel

Figure 21.15: *Primer additions 5' of the template-specific part of the primer.*

At the top of the dialog (see figure 21.15), you can specify primer additions such as a Shine-Dalgarno site, start codon etc. Click in the text field and press **Shift + F1** to show some of the most common additions (see figure 21.16).

Use the up and down arrow keys to select a tag and press **Enter**. This will insert the selected sequence as shown in figure 21.17.

At the bottom of the dialog, you can see a preview of what the final PCR product will look like. In the middle there is the sequence of interest (i.e. the sequence you selected as input). In the beginning is the attB1 site, and at the end is the attB2 site. The primer additions that you have inserted are shown in colors (like the green Shine-Dalgarno site in figure 21.17).

This default list of primer additions can be modified, see section 21.2.1.

You can also manually type a sequence with the keyboard or paste in a sequence from the clipboard by pressing **Ctrl + v (⌘ + v on Mac)**.

Clicking **Next** allows you to specify the length of the template-specific part of the primers as shown in figure 21.18.

The *CLC Genomics Workbench* is not doing any kind of primer design when adding the attB sites. As a user, you simply specify the length of the template-specific part of the primer, and together with the attB sites and optional primer additions, this will be the primer. The primer region will be annotated in the resulting attB-flanked sequence and you can also get a list of primers as you

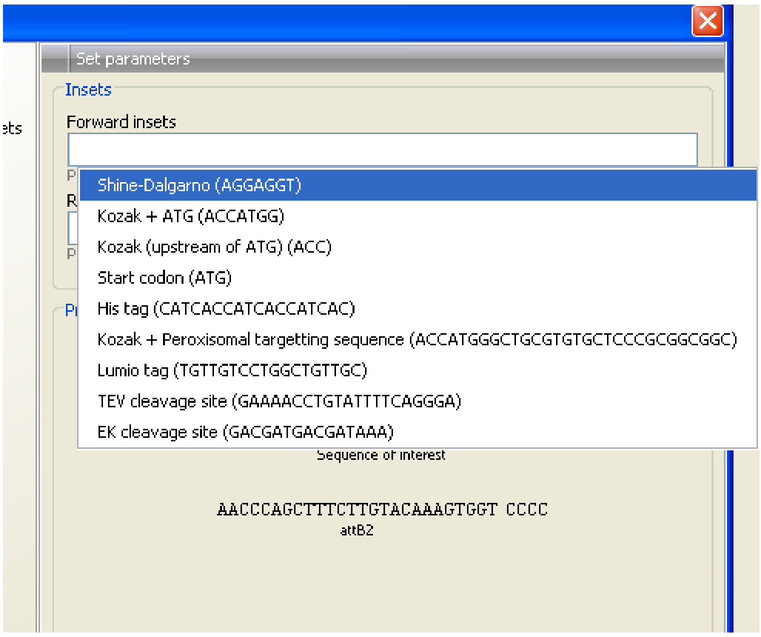
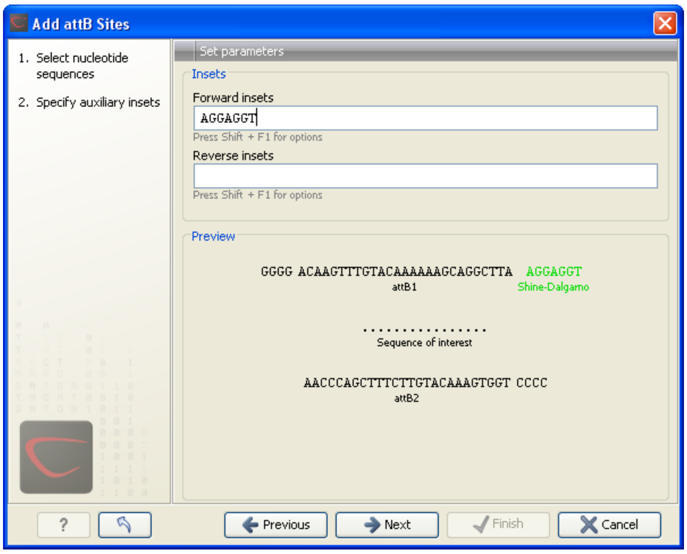

Set parameters
Insets
Forward insets
Shine-Dalgarno (AGGAGGT)
Kozak + ATG (ACCATGG)
Kozak (upstream of ATG) (ACC)
Start codon (ATG)
His tag (CATCACCATCACCATCAC)
Kozak + Peroxisomal targetting sequence (ACCATGGGCTGCGTGTGCTCCCGCGGCGGC)
Lumio tag (TGTTGTCCTGGCTGTTGC)
TEV cleavage site (GAAAACCTGTATTTTCAGGGA)
EK cleavage site (GACGATGACGATAAA)
Sequence of interest
AACCCAGCTTTCTTGTACAAAGTGGT CCCC
attB2

Figure 21.16: *Pressing Shift + F1 shows some of the common additions. This default list can be modified, see section 21.2.1.*


Add attB Sites
1. Select nucleotide sequences
2. Specify auxiliary insets
Set parameters
Insets
Preview
GGGG ACAAGTTTGTACAAAAAAGCAGGCTTA AGGAGGT
attB1
Shine-Dalgarno
....
Sequence of interest
AACCCAGCTTTCTTGTACAAAGTGGT CCCC
attB2
?
Previous
Next
Finish
Cancel

Forward insets: AGGAGGT

Press Shift + F1 for options

Reverse insets:

Press Shift + F1 for options

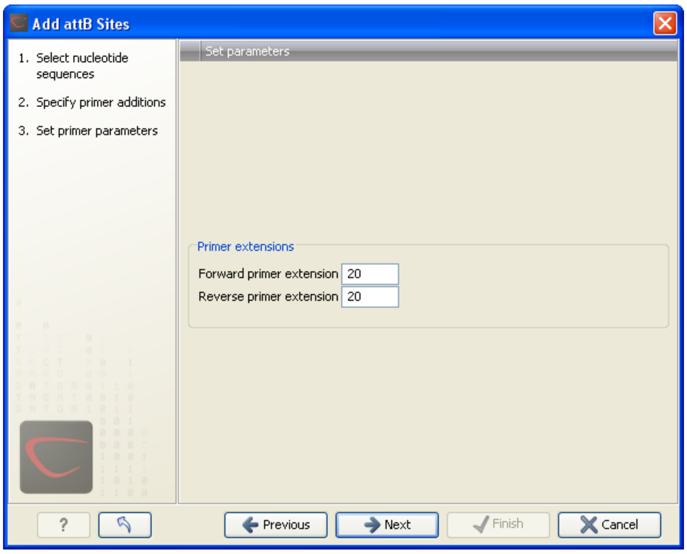
Figure 21.17: *A Shine-Dalgarno sequence has been inserted.*

can see when clicking **Next** (see figure 21.19.

Besides the main output which is a copy of the the input sequence(s) now including attB sites and primer additions, you can get a list of primers as output. Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

The attB sites, the primer additions and the primer regions are annotated in the final result as shown in figure 21.20.

There will be one output sequence for each sequence you have selected for adding attB sites. **Save** (💾) the resulting sequence as it will be the input to the next part of the Gateway cloning work flow (see section 21.2.2). When you open the sequence again, you may need to switch on the relevant annotation types to show the sites and primer additions as illustrated in figure



Add attB Sites

1. Select nucleotide sequences
2. Specify primer additions
3. Set primer parameters

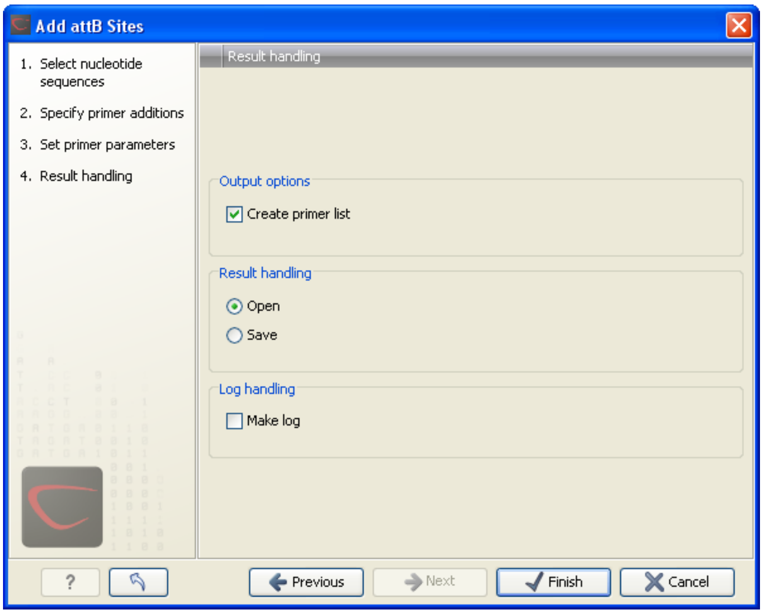
Set parameters

Primer extensions

Forward primer extension 20

Reverse primer extension 20

Figure 21.18: *Specifying the length of the template-specific part of the primers.*



Add attB Sites

1. Select nucleotide sequences
2. Specify primer additions
3. Set primer parameters
4. Result handling

Result handling

Output options

☑ Create primer list

Result handling

● Open

○ Save

Log handling

□ Make log

Figure 21.19: *Besides the main output which is a copy of the the input sequence(s) now including attB sites and primer additions, you can get a list of primers as output.*

21.20.

Extending the pre-defined list of primer additions

The list of primer additions shown when pressing **Shift+F1** in the dialog shown in figure 21.15 can be configured and extended. If there is a tag that you use a lot, you can add it to the list for convenient and easy access later on. This is done in the **Preferences**:

Edit | **Preferences** | **Advanced**

In the advanced preferences dialog, scroll to the part called **Gateway cloning primer additions** (see figure 21.21).

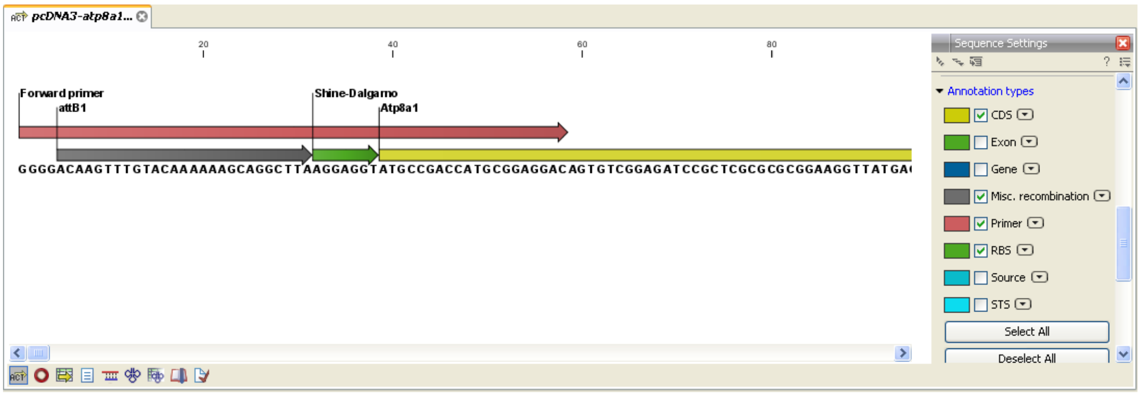
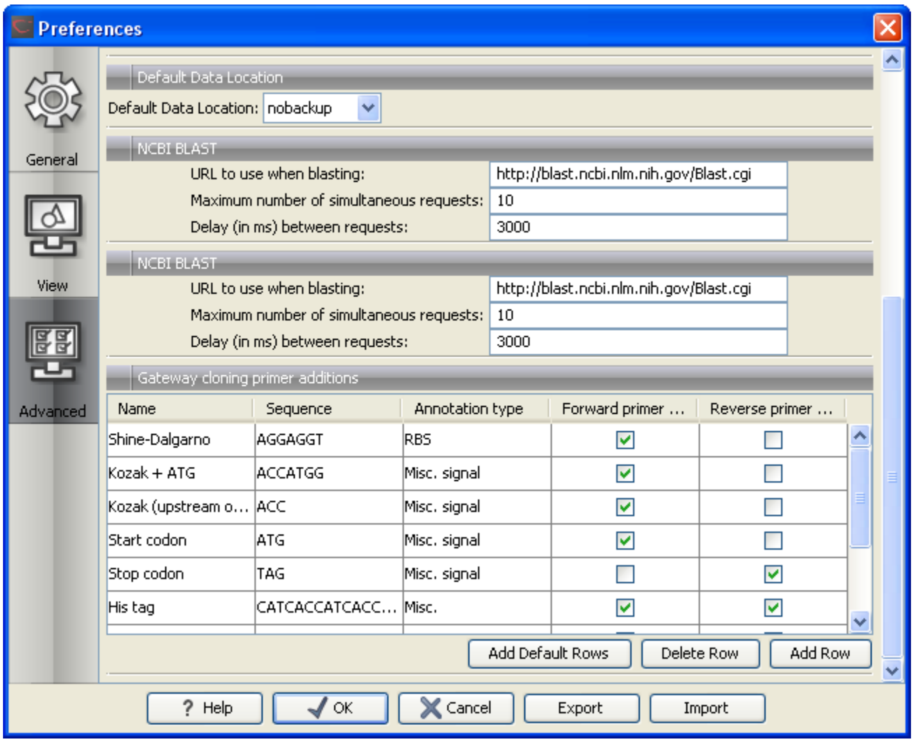


Figure 21.20: *the attB site plus the Shine-Dalgarno primer addition is annotated.*



Default Data Location

Default Data Location: nobackup

NCBI BLAST

URL to use when blasting: http://blast.ncbi.nlm.nih.gov/Blast.cgi

Maximum number of simultaneous requests: 10

Delay (in ms) between requests: 3000

NCBI BLAST

URL to use when blasting: http://blast.ncbi.nlm.nih.gov/Blast.cgi

Maximum number of simultaneous requests: 10

Delay (in ms) between requests: 3000

Gateway cloning primer additions

Name	Sequence	Annotation type	Forward primer ...	Reverse primer ...
Shine-Dalgarno	AGGAGGT	RBS	☑	☐
Kozak + ATG	ACCATGG	Misc. signal	☑	☐
Kozak (upstream o...	ACC	Misc. signal	☑	☐
Start codon	ATG	Misc. signal	☑	☐
Stop codon	TAG	Misc. signal	☐	☑
His tag	CATCACCATCACC...	Misc.	☑	☑

Figure 21.21: *Configuring the list of primer additions available when adding attB sites.*

Each element in the list has the following information:

Name The name of the sequence. When the sequence fragment is extended with a primer addition, an annotation will be added displaying this name.

Sequence The actual sequence to be inserted. The sequence is always defined on the sense strand (although the reverse primer would be reverse complement).

Annotation type The annotation type used for the annotation that is added to the fragment.

Forward primer addition Whether this addition should be visible in the list of additions for the forward primer.

Reverse primer addition Whether this addition should be visible in the list of additions for the reverse primer.

You can either change the existing elements in the table by double-clicking any of the cells, or you can use the buttons below to: **Add Row** or **Delete Row**. If you by accident have deleted or modified some of the default primer additions, you can press **Add Default Rows**. Note that this will not reset the table but only add all the default rows to the existing rows.

21.2.2 Create entry clones (BP)

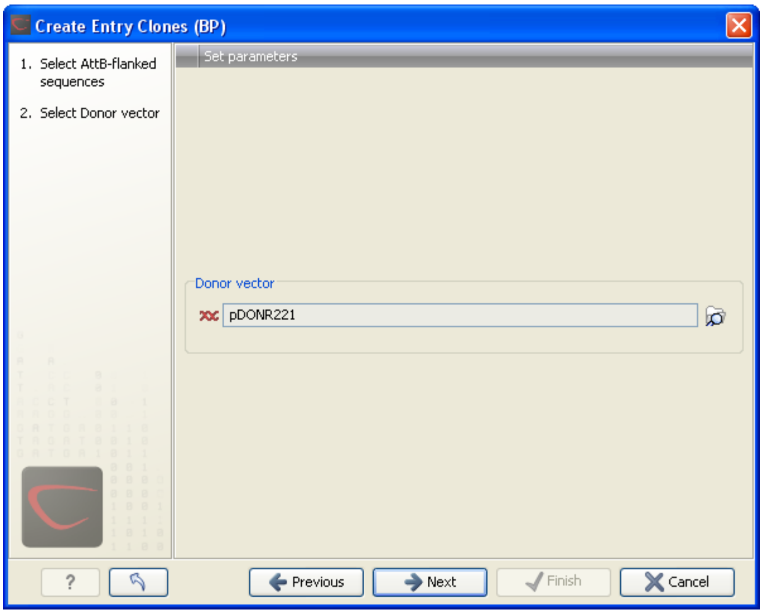
The next step in the Gateway cloning work flow is to recombine the attB-flanked sequence of interest into a donor vector to create an entry clone, the so-called BP reaction:

> **Toolbox in the Menu Bar | Cloning and Restriction Sites ( ) | Gateway Cloning ( ) | Create Entry Clone (BP) ( )**

This will open a dialog where you can select on ore more sequences that will be the sequence of interest to be recombined into your donor vector. Note that the sequences you select should be flanked with attB sites (see section 21.2.1). You can select more than one sequence as input, and the corresponding number of entry clones will be created.

When you have selected your sequence(s), click **Next**.

This will display the dialog shown in figure 21.22.



Donor vector: pDONR221

Figure 21.22: *Selecting one or more donor vectors.*

Clicking the **Browse** ( ) button opens a dialog where you can select a donor vector. You can download donor vectors from Invitrogen's web site: `http://tools.invitrogen.com/downloads/Gateway%20vectors.ma4` and import into the *CLC Genomics Workbench*. Note that the Workbench looks for the specific sequences of the attP sites in the sequences that you select in this dialog (see how to change the definition of sites in section G).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

The output is one entry clone per sequence selected. The attB and attP sites have been used for the recombination, and the entry clone is now equipped with attL sites as shown in figure 21.23.

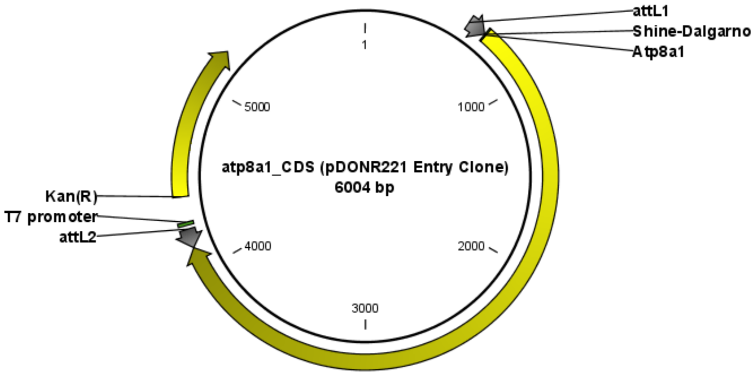


Figure 21.23: *The resulting entry vector opened in a circular view.*

Note that the bi-product of the recombination is not part of the output.

21.2.3 Create expression clones (LR)

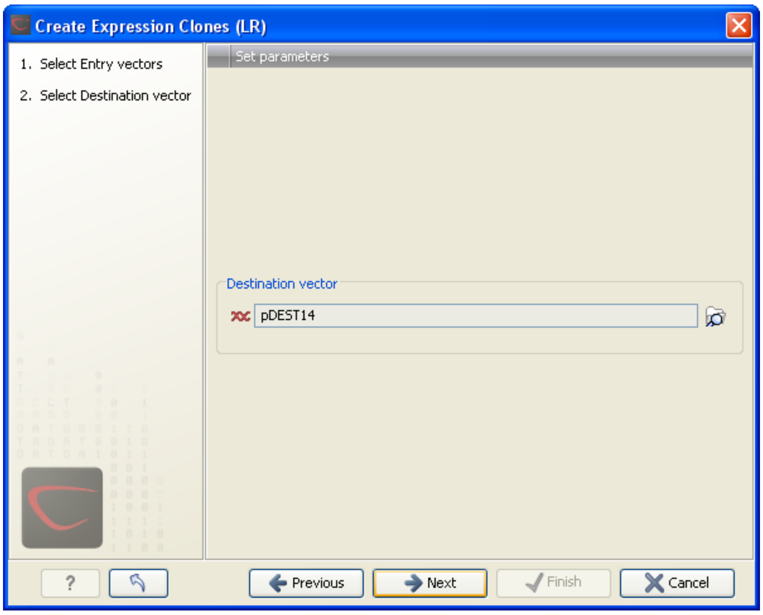
The final step in the Gateway cloning work flow is to recombine the entry clone into a donor vector to create an expression clone, the so-called LR reaction:

> **Toolbox in the Menu Bar | Cloning and Restriction Sites** () **| Gateway Cloning** () **| Create Expression Clone (LR)** ()

This will open a dialog where you can select on ore more entry clones (see how to create an entry clone in section 21.2.2). Note that the entry clone sequences you select should contain attL sites (see how to change the definition of sites in section G). You can select more than one entry clone as input, and the corresponding number of expression clones will be created.

When you have selected your entry clone(s), click **Next**.

This will display the dialog shown in figure 21.24.



Destination vector: pDEST14

Figure 21.24: *Selecting one or more destination vectors.*

Clicking the **Browse** (🔍) button opens a dialog where you can select a destination vector. You can download donor vectors from Invitrogen's web site: http://tools.invitrogen.com/downloads/Gateway%20vectors.ma4 and import into the *CLC Genomics Workbench*. Note that the Workbench looks for the specific sequences of the attR sites in the sequences that you select in this dialog (see how to change the definition of sites in section G).

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

The output is a number of expression clones depending on how many entry clones and destination vectors that you selected. The attL and attR sites have been used for the recombination, and the expression clone is now equipped with attB sites as shown in figure 21.25.

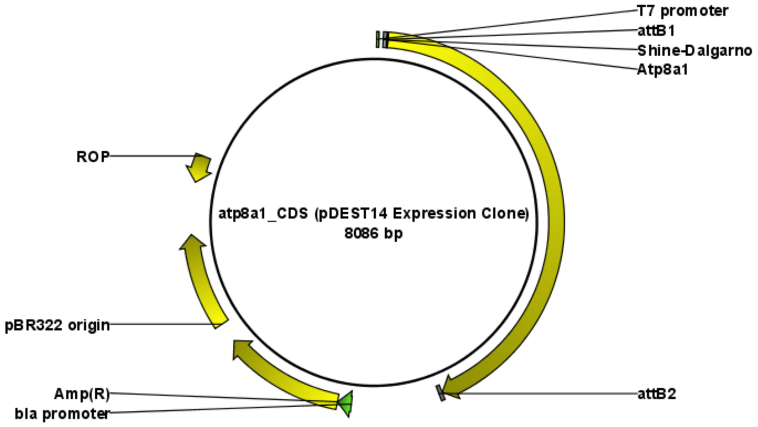


Figure 21.25: *The resulting expression clone opened in a circular view.*

Note that the bi-product of the recombination is not part of the output.

21.3 Restriction site analysis

There are two ways of finding and showing restriction sites:

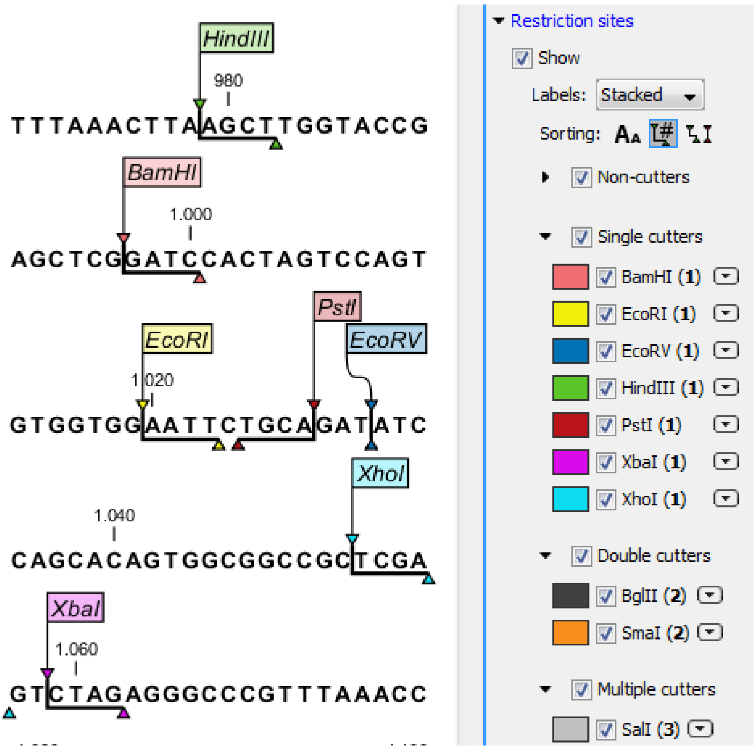
- In many cases, the dynamic restriction sites found in the **Side Panel** of sequence views will be useful, since it is a quick and easy way of showing restriction sites.
- In the **Toolbox** you will find the other way of doing restriction site analyses. This way provides more control of the analysis and gives you more output options, e.g. a table of restriction sites and you can perform the same restriction map analysis on several sequences in one step.

This chapter first describes the dynamic restriction sites, followed by "the toolbox way". This section also includes an explanation of how to simulate a gel with the selected enzymes. The final section in this chapter focuses on enzyme lists which represent an easy way of managing restriction enzymes.

21.3.1 Dynamic restriction sites

If you open a sequence, a sequence list etc, you will find the **Restriction Sites** group in the **Side Panel**.

As shown in figure 21.26 you can display restriction sites as colored triangles and lines on the sequence. The **Restriction sites** group in the side panel shows a list of enzymes, represented by different colors corresponding to the colors of the triangles on the sequence. By selecting or deselecting the enzymes in the list, you can specify which enzymes' restriction sites should be displayed.



▾ Restriction sites

☑ Show

Labels: Stacked ▾

Sorting: Aᴀ ⌗ ⇅

▸ ☑ Non-cutters

▾ ☑ Single cutters

🟥 ☑ BamHI (1)
🟨 ☑ EcoRI (1)
🟦 ☑ EcoRV (1)
🟩 ☑ HindIII (1)
🟥 ☑ PstI (1)
🟪 ☑ XbaI (1)
🟦 ☑ XhoI (1)

▾ ☑ Double cutters

⬛ ☑ BglII (2)
🟧 ☑ SmaI (2)

▾ ☑ Multiple cutters

⬜ ☑ SalI (3)

Figure 21.26: *Showing restriction sites of ten restriction enzymes.*

The color of the restriction enzyme can be changed by clicking the colored box next to the enzyme's name. The name of the enzyme can also be shown next to the restriction site by selecting **Show name flags** above the list of restriction enzymes.

There is also an option to specify how the **Labels** shown be shown:

- **No labels**. This will just display the cut site with no information about the name of the enzyme. Placing the mouse button on the cut site will reveal this information as a tool tip.
- **Flag**. This will place a flag just above the sequence with the enzyme name (see an example in figure 21.27). Note that this option will make it hard to see when several cut sites are located close to each other. In the circular view, this option is replaced by the Radial option:
- **Radial**. This option is only available in the circular view. It will place the restriction site labels as close to the cut site as possible (see an example in figure 21.29).
- **Stacked**. This is similar to the flag option for linear sequence views, but it will stack the labels so that all enzymes are shown. For circular views, it will align all the labels on each side of the circle. This can be useful for clearly seeing the order of the cut sites when they are located closely together (see an example in figure 21.28).

Note that in a circular view, the **Stacked** and **Radial** options also affect the layout of annotations.

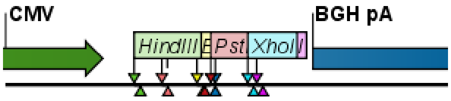


Figure 21.27: *Restriction site labels shown as flags.*

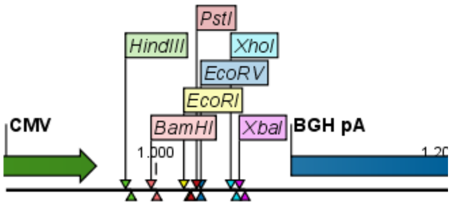


Figure 21.28: *Restriction site labels stacked.*

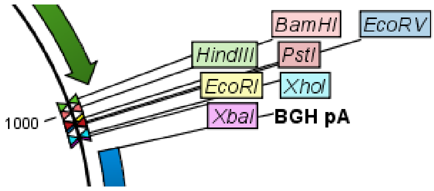


Figure 21.29: *Restriction site labels in radial layout.*

Sort enzymes

Just above the list of enzymes there are three buttons to be used for sorting the list (see figure 21.30):

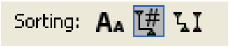


Figure 21.30: *Buttons to sort restriction enzymes.*

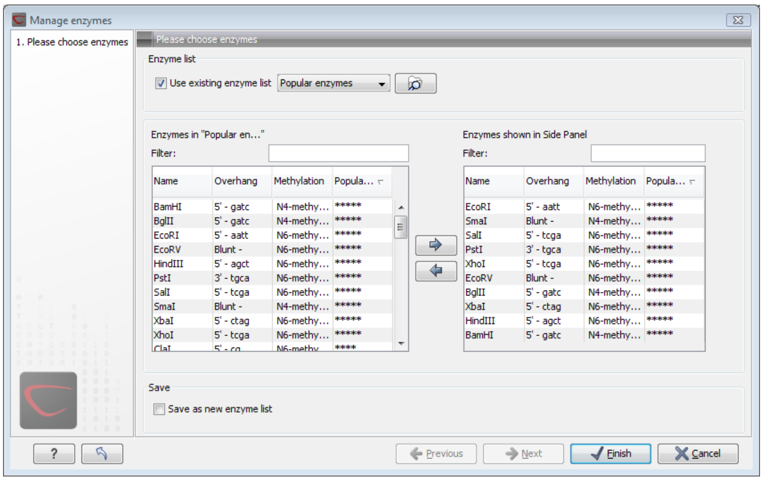
- **Sort enzymes alphabetically** (AA). Clicking this button will sort the list of enzymes alphabetically.
- **Sort enzymes by number of restriction sites** (#). This will divide the enzymes into four groups:
  - Non-cutters.
  - Single cutters.
  - Double cutters.
  - Multiple cutters.

  There is a checkbox for each group which can be used to hide / show all the enzymes in a group.
-
- **Sort enzymes by overhang** (⇃I). This will divide the enzymes into three groups:
  - Blunt. Enzymes cutting both strands at the same position.
  - 3'. Enzymes producing an overhang at the 3' end.
  - 5'. Enzymes producing an overhang at the 5' end.

  There is a checkbox for each group which can be used to hide / show all the enzymes in a group.

Manage enzymes

The list of restriction enzymes contains per default 20 of the most popular enzymes, but you can easily modify this list and add more enzymes by clicking the **Manage enzymes button.** This will display the dialog shown in figure 21.31.



Enzyme list

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter:

Name	Overhang	Methylation	Popula...
BamHI	5' - gatc	N4-methy...	*****
BglII	5' - gatc	N4-methy...	*****
EcoRI	5' - aatt	N6-methy...	*****
EcoRV	Blunt -	N6-methy...	*****
HindIII	5' - agct	N6-methy...	*****
PstI	3' - tgca	N6-methy...	*****
SalI	5' - tcga	N6-methy...	*****
SmaI	Blunt -	N4-methy...	*****
XbaI	5' - ctag	N6-methy...	*****
XhoI	5' - tcga	N6-methy...	*****
ClaI	5' - cg	N6-methy	****

Enzymes shown in Side Panel

Filter:

Name	Overhang	Methylation	Popula...
EcoRI	5' - aatt	N6-methy...	*****
SmaI	Blunt -	N4-methy...	*****
SalI	5' - tcga	N6-methy...	*****
PstI	3' - tgca	N6-methy...	*****
XhoI	5' - tcga	N6-methy...	*****
EcoRV	Blunt -	N6-methy...	*****
BglII	5' - gatc	N4-methy...	*****
XbaI	5' - ctag	N6-methy...	*****
HindIII	5' - agct	N6-methy...	*****
BamHI	5' - gatc	N4-methy...	*****

Save

□ Save as new enzyme list

Figure 21.31: *Adding or removing enzymes from the Side Panel.*

At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. See section 21.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you see all the enzymes that are in the list select above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [3].
- To the **right**, there is a list of the enzymes that will be used.

Select enzymes in the left side panel and add them to the right panel by double-clicking or clicking the **Add** button (➡). If you e.g. wish to use EcoRV and BamHI, select these two enzymes and add them to the right side panel.

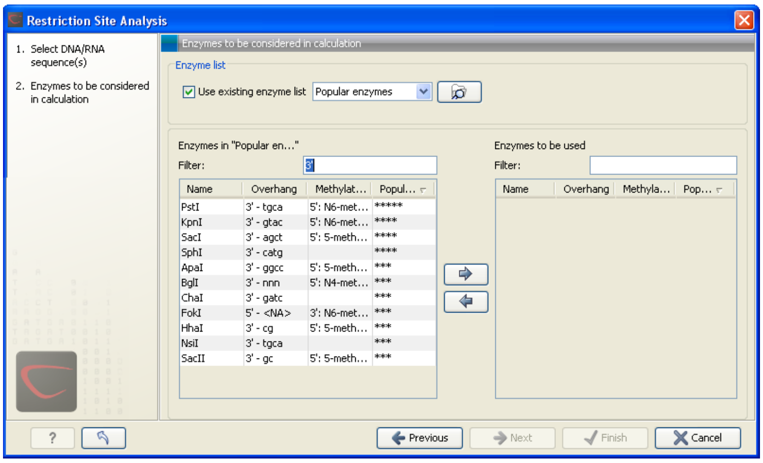
If you wish to use all the enzymes in the list:

Click in the panel to the left | press Ctrl + A (⌘ + A on Mac) | Add (➡)

The enzymes can be sorted by clicking the column headings, i.e. Name, Overhang, Methylation or Popularity. This is particularly useful if you wish to use enzymes which produce e.g. a 3' overhang. In this case, you can sort the list by clicking the Overhang column heading, and all the enzymes producing 3' overhangs will be listed together for easy selection.

When looking for a specific enzyme, it is easier to use the Filter. If you wish to find e.g. HindIII sites, simply type HindIII into the filter, and the list of enzymes will shrink automatically to only include the HindIII enzyme. This can also be used to only show enzymes producing e.g. a 3' overhang as shown in figure 21.50.

[3] The *CLC Genomics Workbench* comes with a standard set of enzymes based on `http://www.rebase.neb.com`. You can customize the enzyme database for your installation, see section F



Restriction Site Analysis

1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation

Enzymes to be considered in calculation

Enzyme list

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter: 3'

Name	Overhang	Methylat...	Popul...
PstI	3' - tgca	5': N6-met...	*****
KpnI	3' - gtac	5': N6-met...	****
SacI	3' - agct	5': 5-meth...	****
SphI	3' - catg		****
ApaI	3' - ggcc	5': 5-meth...	***
BglI	3' - nnn	5': N4-met...	***
ChaI	3' - gatc		***
FokI	5' - <NA>	3': N6-met...	***
HhaI	3' - cg	5': 5-meth...	***
NsiI	3' - tgca		***
SacII	3' - gc	5': 5-meth...	***

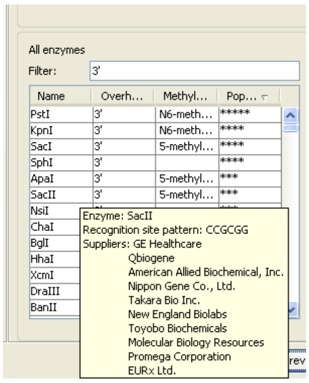
Enzymes to be used

Filter:

Name	Overhang	Methyla...	Pop...

Figure 21.32: *Selecting enzymes.*

If you need more detailed information and filtering of the enzymes, either place your mouse cursor on an enzyme for one second to display additional information (see figure 21.51), or use the view of enzyme lists (see 21.5).



All enzymes

Filter: 3'

Name	Overh...	Methyl...	Pop...
PstI	3'	N6-meth...	*****
KpnI	3'	N6-meth...	****
SacI	3'	5-methyl...	****
SphI	3'		****
ApaI	3'	5-methyl...	***
SacII	3'	5-methyl...	***
NsiI			
ChaI			
BglI			
HhaI			
XcmI			
DraIII			
BanII			

Enzyme: SacII
Recognition site pattern: CCGCGG
Suppliers: GE Healthcare
Qbiogene
American Allied Biochemical, Inc.
Nippon Gene Co., Ltd.
Takara Bio Inc.
New England Biolabs
Toyobo Biochemicals
Molecular Biology Resources
Promega Corporation
EURx Ltd.

Figure 21.33: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

At the bottom of the dialog, you can select to save this list of enzymes as a new file. In this way, you can save the selection of enzymes for later use.

When you click **Finish**, the enzymes are added to the Side Panel and the cut sites are shown on the sequence.

If you have specified a set of enzymes which you always use, it will probably be a good idea to save the settings in the Side Panel (see section 3.2.7) for future use.

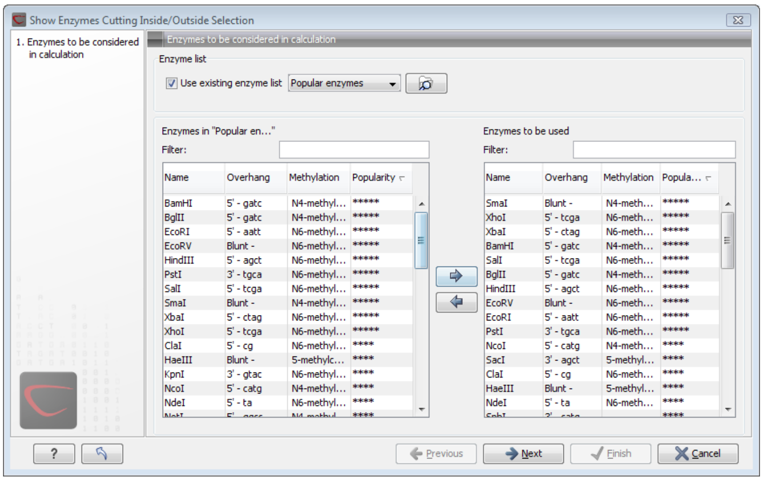
Show enzymes cutting inside/outside selection

Section 21.3.1 describes how to add more enzymes to the list in the Side Panel based on the name of the enzyme, overhang, methylation sensitivity etc. However, you will often find yourself in a situation where you need a more sophisticated and explorative approach.

An illustrative example: you have a selection on a sequence, and you wish to find enzymes cutting within the selection, but not outside. This problem often arises during design of cloning experiments. In this case, you do not know the name of the enzyme, so you want the Workbench to find the enzymes for you:

right-click the selection | Show Enzymes Cutting Inside/Outside Selection ( )

This will display the dialog shown in figure 21.34 where you can specify which enzymes should initially be considered.



Show Enzymes Cutting Inside/Outside Selection

1. Enzymes to be considered in calculation

Enzymes to be considered in calculation

Enzyme list

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter:

Name	Overhang	Methylation	Popularity
BamHI	5' - gatc	N4-methyl...	*****
BglII	5' - gatc	N4-methyl...	*****
EcoRI	5' - aatt	N6-methyl...	*****
EcoRV	Blunt -	N6-methyl...	*****
HindIII	5' - agct	N6-methyl...	*****
PstI	3' - tgca	N6-methyl...	*****
SalI	5' - tcga	N6-methyl...	*****
SmaI	Blunt -	N4-methyl...	*****
XbaI	5' - ctag	N6-methyl...	*****
XhoI	5' - tcga	N6-methyl...	*****
ClaI	5' - cg	N6-methyl...	****
HaeIII	Blunt -	5-methylc...	****
KpnI	3' - gtac	N6-methyl...	****
NcoI	5' - catg	N4-methyl...	****
NdeI	5' - ta	N6-methyl...	****

Enzymes to be used

Filter:

Name	Overhang	Methylation	Popula...
SmaI	Blunt -	N4-meth...	*****
XhoI	5' - tcga	N6-meth...	*****
XbaI	5' - ctag	N6-meth...	*****
BamHI	5' - gatc	N4-meth...	*****
SalI	5' - tcga	N6-meth...	*****
BglII	5' - gatc	N4-meth...	*****
HindIII	5' - agct	N6-meth...	*****
EcoRV	Blunt -	N6-meth...	*****
EcoRI	5' - aatt	N6-meth...	*****
PstI	3' - tgca	N6-meth...	*****
NcoI	5' - catg	N4-meth...	****
SacI	3' - agct	5-methyl...	****
ClaI	5' - cg	N6-meth...	****
HaeIII	Blunt -	5-methyl...	****
NdeI	5' - ta	N6-meth...	****

Figure 21.34: *Choosing enzymes to be considered.*

At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. See section 21.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you see all the enzymes that are in the list select above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [4].
- To the **right**, there is a list of the enzymes that will be used.

Select enzymes in the left side panel and add them to the right panel by double-clicking or clicking the **Add** button ( ). If you e.g. wish to use EcoRV and BamHI, select these two enzymes and add them to the right side panel.

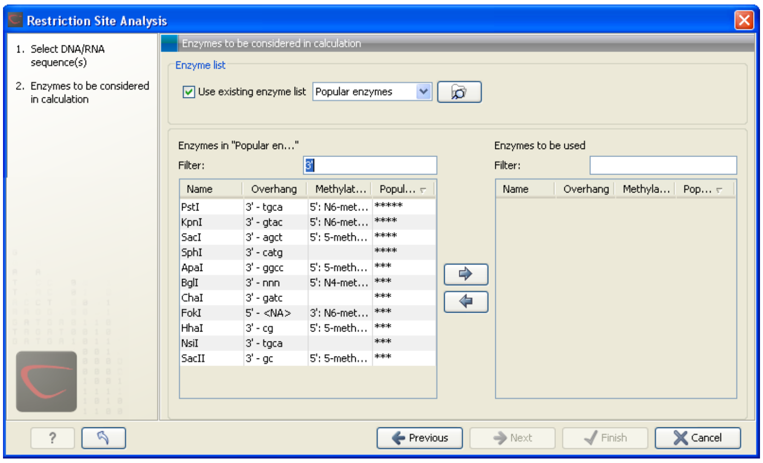
If you wish to use all the enzymes in the list:

Click in the panel to the left | press Ctrl + A (⌘ + A on Mac) | Add ( )

The enzymes can be sorted by clicking the column headings, i.e. Name, Overhang, Methylation or Popularity. This is particularly useful if you wish to use enzymes which produce e.g. a 3' overhang. In this case, you can sort the list by clicking the Overhang column heading, and all the enzymes producing 3' overhangs will be listed together for easy selection.

When looking for a specific enzyme, it is easier to use the Filter. If you wish to find e.g. HindIII sites, simply type HindIII into the filter, and the list of enzymes will shrink automatically to only include the HindIII enzyme. This can also be used to only show enzymes producing e.g. a 3' overhang as shown in figure 21.50.

[4] The *CLC Genomics Workbench* comes with a standard set of enzymes based on http://www.rebase.neb.com. You can customize the enzyme database for your installation, see section F



1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation

Enzymes to be considered in calculation

Enzyme list

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter: 3'

Name	Overhang	Methylat...	Popul...
PstI	3' - tgca	5': N6-met...	*****
KpnI	3' - gtac	5': N6-met...	****
SacI	3' - agct	5': 5-meth...	****
SphI	3' - catg		****
ApaI	3' - ggcc	5': 5-meth...	***
BglI	3' - nnn	5': N4-met...	***
ChaI	3' - gatc		***
FokI	5' - <NA>	3': N6-met...	***
HhaI	3' - cg	5': 5-meth...	***
NsiI	3' - tgca		***
SacII	3' - gc	5': 5-meth...	***

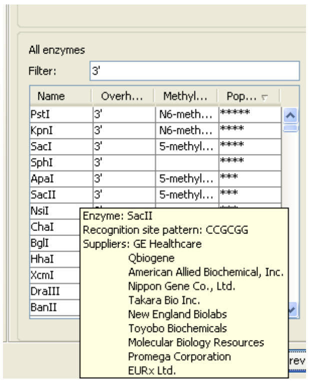
Enzymes to be used

Filter:

Name	Overhang	Methyla...	Pop...

Figure 21.35: *Selecting enzymes.*

If you need more detailed information and filtering of the enzymes, either place your mouse cursor on an enzyme for one second to display additional information (see figure 21.51), or use the view of enzyme lists (see 21.5).



All enzymes

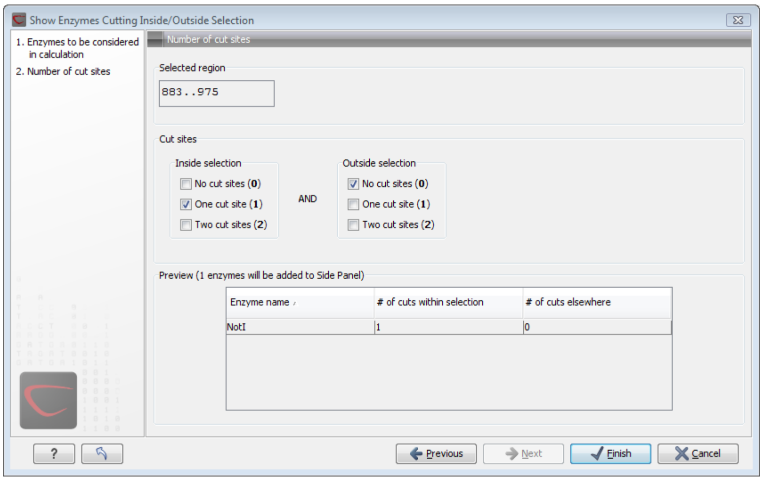
Filter: 3'

Name	Overh...	Methyl...	Pop...
PstI	3'	N6-meth...	*****
KpnI	3'	N6-meth...	****
SacI	3'	5-methyl...	****
SphI	3'		****
ApaI	3'	5-methyl...	***
SacII	3'	5-methyl...	***
NsiI			
ChaI			
BglI			
HhaI			
XcmI			
DraIII			
BanII			

Enzyme: SacII
Recognition site pattern: CCGCGG
Suppliers: GE Healthcare
Qbiogene
American Allied Biochemical, Inc.
Nippon Gene Co., Ltd.
Takara Bio Inc.
New England Biolabs
Toyobo Biochemicals
Molecular Biology Resources
Promega Corporation
EURx Ltd.

Figure 21.36: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

Clicking **Next** will show the dialog in figure 21.37.



1. Enzymes to be considered in calculation
2. Number of cut sites

Number of cut sites

Selected region: 883..975

Cut sites

Inside selection
□ No cut sites (0)
☑ One cut site (1)
□ Two cut sites (2)

AND

Outside selection
☑ No cut sites (0)
□ One cut site (1)
□ Two cut sites (2)

Preview (1 enzymes will be added to Side Panel)

Enzyme name	# of cuts within selection	# of cuts elsewhere
NotI	1	0

Figure 21.37: *Deciding number of cut sites inside and outside the selection.*

At the top of the dialog, you see the selected region, and below are two panels:

- **Inside selection**. Specify how many times you wish the enzyme to cut inside the selection. In the example described above, "One cut site (1)" should be selected to only show enzymes cutting once in the selection.

- **Outside selection**. Specify how many times you wish the enzyme to cut outside the selection (i.e. the rest of the sequence). In the example above, "No cut sites (0)" should be selected.

These panels offer a lot of flexibility for combining number of cut sites inside and outside the selection, respectively. To give a hint of how many enzymes will be added based on the combination of cut sites, the preview panel at the bottom lists the enzymes which will be added when you click **Finish**. Note that this list is dynamically updated when you change the number of cut sites. The enzymes shown in brackets [] are enzymes which are already present in the Side Panel.

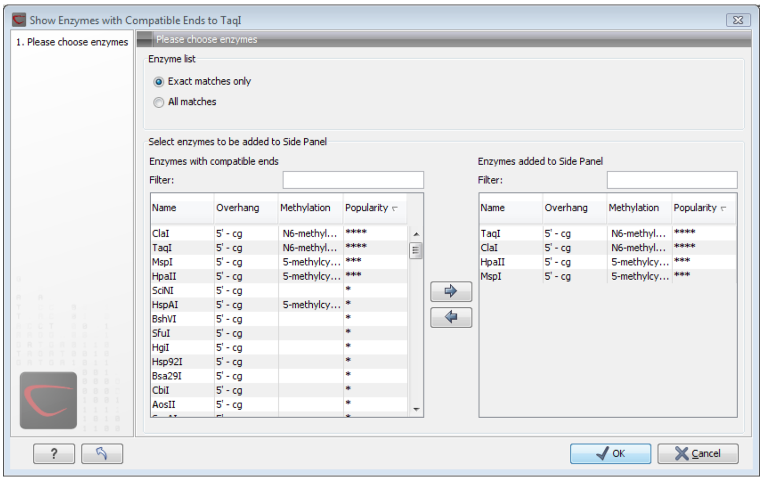
If you have selected more than one region on the sequence (using Ctrl or ⌘), they will be treated as individual regions. This means that the criteria for cut sites apply to each region.

Show enzymes with compatible ends

Besides what is described above, there is a third way of adding enzymes to the Side Panel and thereby displaying them on the sequence. It is based on the overhang produced by cutting with an enzyme and will find enzymes producing a compatible overhang:

right-click the restriction site | Show Enzymes with Compatible Ends (↴I)

This will display the dialog shown in figure 21.38.



Show Enzymes with Compatible Ends to TaqI

1. Please choose enzymes

Please choose enzymes

Enzyme list

● Exact matches only

○ All matches

Select enzymes to be added to Side Panel

Enzymes with compatible ends

Filter:

Name	Overhang	Methylation	Popularity
ClaI	5' - cg	N6-methyl...	****
TaqI	5' - cg	N6-methyl...	****
MspI	5' - cg	5-methylcy...	***
HpaII	5' - cg	5-methylcy...	***
SciNI	5' - cg		*
HspAI	5' - cg	5-methylcy...	*
BshVI	5' - cg		*
SfuI	5' - cg		*
HgiI	5' - cg		*
Hsp92I	5' - cg		*
Bsa29I	5' - cg		*
CbiI	5' - cg		*
AosII	5' - cg		*

Enzymes added to Side Panel

Filter:

Name	Overhang	Methylation	Popularity
TaqI	5' - cg	N6-methyl...	****
ClaI	5' - cg	N6-methyl...	****
HpaII	5' - cg	5-methylcy...	***
MspI	5' - cg	5-methylcy...	***

Figure 21.38: *Enzymes with compatible ends.*

At the top you can choose whether the enzymes considered should have an exact match or not. Since a number of restriction enzymes have ambiguous cut patterns, there will be variations in the resulting overhangs. Choosing **All matches**, you cannot be 100% sure that the overhang will match, and you will need to inspect the sequence further afterwards.

We advice trying **Exact match** first, and use **All matches** as an alternative if a satisfactory result cannot be achieved.

At the bottom of the dialog, the list of enzymes producing compatible overhangs is shown. Use the arrows to add enzymes which will be displayed on the sequence which you press **Finish**.

When you have added the relevant enzymes, click **Finish**, and the enzymes will be added to the Side Panel and their cut sites displayed on the sequence.

21.3.2 Restriction site analysis from the Toolbox

Besides the dynamic restriction sites, you can do a more elaborate restriction map analysis with more output format using the Toolbox:

Toolbox | **Cloning and Restriction Sites** () | **Restriction Site Analysis** ()

This will display the dialog shown in figure 21.39.

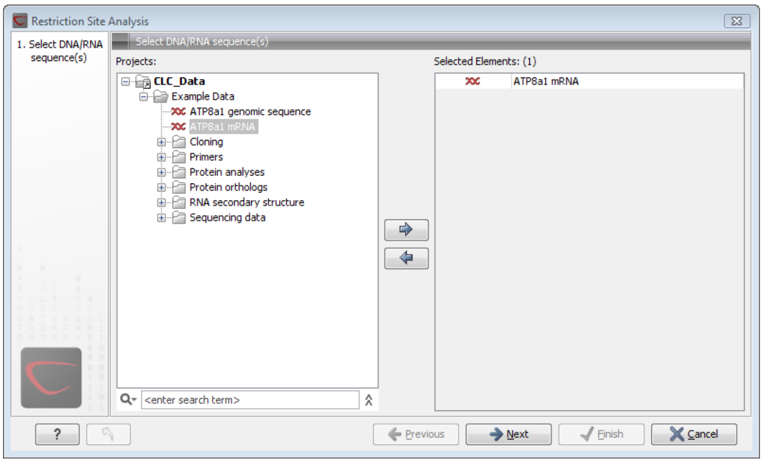


Figure 21.39: *Choosing sequence ATP8a1 mRNA for restriction map analysis.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Selecting, sorting and filtering enzymes

Clicking **Next** lets you define which enzymes to use as basis for finding restriction sites on the sequence. At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. See section 21.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you see all the enzymes that are in the list select above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [5].
- To the **right**, there is a list of the enzymes that will be used.

[5] The *CLC Genomics Workbench* comes with a standard set of enzymes based on `http://www.rebase.neb.com`. You can customize the enzyme database for your installation, see section F

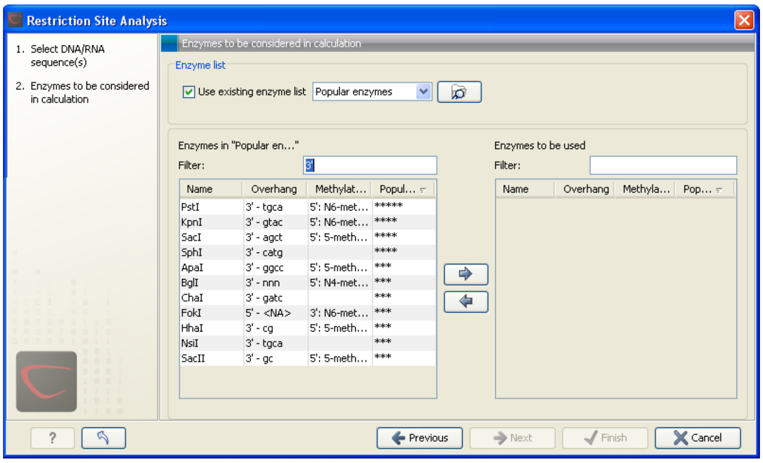
Select enzymes in the left side panel and add them to the right panel by double-clicking or clicking the **Add** button (➡). If you e.g. wish to use EcoRV and BamHI, select these two enzymes and add them to the right side panel.

If you wish to use all the enzymes in the list:

> **Click in the panel to the left | press Ctrl + A (⌘ + A on Mac) | Add (➡)**

The enzymes can be sorted by clicking the column headings, i.e. Name, Overhang, Methylation or Popularity. This is particularly useful if you wish to use enzymes which produce e.g. a 3' overhang. In this case, you can sort the list by clicking the Overhang column heading, and all the enzymes producing 3' overhangs will be listed together for easy selection.

When looking for a specific enzyme, it is easier to use the Filter. If you wish to find e.g. HindIII sites, simply type HindIII into the filter, and the list of enzymes will shrink automatically to only include the HindIII enzyme. This can also be used to only show enzymes producing e.g. a 3' overhang as shown in figure 21.50.


Restriction Site Analysis
Enzymes to be considered in calculation
Enzyme list
➡
⬅
?
Previous
Next
Finish
Cancel

1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter: 3'

Name	Overhang	Methylat...	Popul...
PstI	3' - tgca	5': N6-met...	*****
KpnI	3' - gtac	5': N6-met...	****
SacI	3' - agct	5': 5-meth...	****
SphI	3' - catg		****
ApaI	3' - ggcc	5': 5-meth...	***
BglI	3' - nnn	5': N4-met...	***
ChaI	3' - gatc		***
FokI	5' - <NA>	3': N6-met...	***
HhaI	3' - cg	5': 5-meth...	***
NsiI	3' - tgca		***
SacII	3' - gc	5': 5-meth...	***

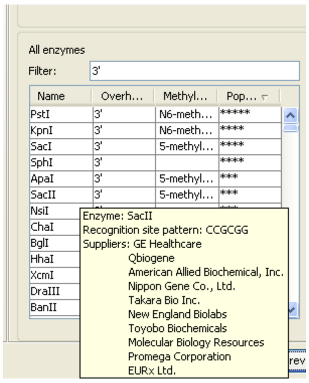
Enzymes to be used

Filter:

Name	Overhang	Methyla...	Pop...

Figure 21.40: *Selecting enzymes.*

If you need more detailed information and filtering of the enzymes, either place your mouse cursor on an enzyme for one second to display additional information (see figure 21.51), or use the view of enzyme lists (see 21.5).



All enzymes

Filter: 3'

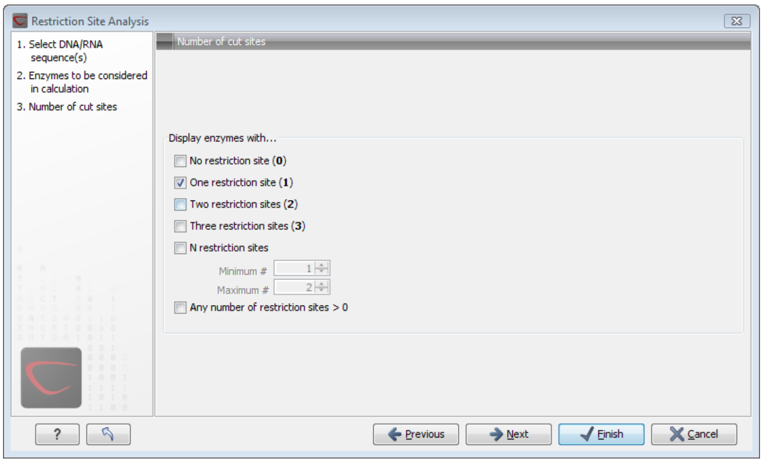
Name	Overh...	Methyl...	Pop...
PstI	3'	N6-meth...	*****
KpnI	3'	N6-meth...	****
SacI	3'	5-methyl...	****
SphI	3'		****
ApaI	3'	5-methyl...	***
SacII	3'	5-methyl...	***
NsiI			
ChaI			
BglI			
HhaI			
XcmI			
DraIII			
BanII			

Enzyme: SacII
Recognition site pattern: CCGCGG
Suppliers: GE Healthcare
Qbiogene
American Allied Biochemical, Inc.
Nippon Gene Co., Ltd.
Takara Bio Inc.
New England Biolabs
Toyobo Biochemicals
Molecular Biology Resources
Promega Corporation
EURx Ltd.

Figure 21.41: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

Number of cut sites

Clicking **Next** confirms the list of enzymes which will be included in the analysis, and takes you to the dialog shown in figure 21.42.



1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation
3. Number of cut sites

Display enzymes with...

- ☐ No restriction site (**0**)
- ☑ One restriction site (**1**)
- ☐ Two restriction sites (**2**)
- ☐ Three restriction sites (**3**)
- ☐ N restriction sites
  - Minimum # 1
  - Maximum # 2
- ☐ Any number of restriction sites > 0

Figure 21.42: *Selecting number of cut sites.*

If you wish the output of the restriction map analysis only to include restriction enzymes which cut the sequence a specific number of times, use the checkboxes in this dialog:

- No restriction site (**0**)
- One restriction site (**1**)
- Two restriction sites (**2**)
- Three restriction site (**3**)
- N restriction sites
  - Minimum
  - Maximum
- Any number of restriction sites > 0

The default setting is to include the enzymes which cut the sequence one or two times.

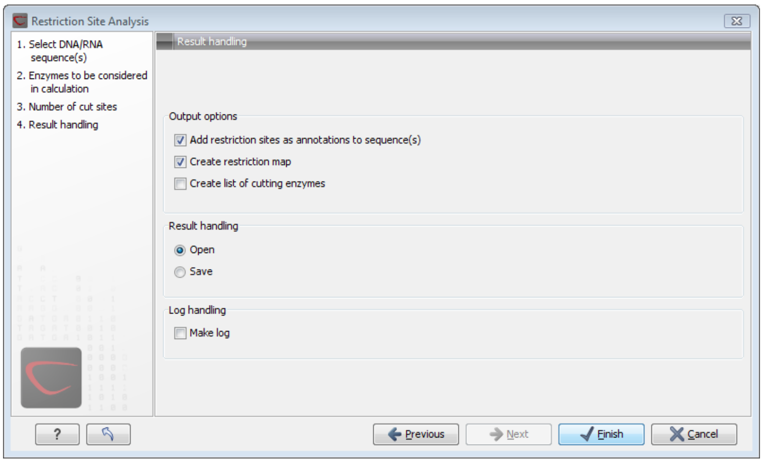
You can use the checkboxes to perform very specific searches for restriction sites: e.g. if you wish to find enzymes which do not cut the sequence, or enzymes cutting exactly twice.

Output of restriction map analysis

Clicking next shows the dialog in figure 21.43.

This dialog lets you specify how the result of the restriction map analysis should be presented:

- **Add restriction sites as annotations to sequence(s)**. This option makes it possible to see the restriction sites on the sequence (see figure 21.44) and save the annotations for later use.



Restriction Site Analysis

1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation
3. Number of cut sites
4. Result handling

Result handling

Output options

☑ Add restriction sites as annotations to sequence(s)

☑ Create restriction map

☐ Create list of cutting enzymes

Result handling

● Open

○ Save

Log handling

☐ Make log

Figure 21.43: *Choosing to add restriction sites as annotations or creating a restriction map.*

- **Create restriction map.** When a restriction map is created, it can be shown in three different ways:
  - As a **table of restriction sites** as shown in figure 21.45. If more than one sequence were selected, the table will include the restriction sites of all the sequences. This makes it easy to compare the result of the restriction map analysis for two sequences.
  - As a **table of fragments** which shows the sequence fragments that would be the result of cutting the sequence with the selected enzymes (see figure21.46).
  - As a **virtual gel** simulation which shows the fragments as bands on a gel (see figure 21.48).

    For more information about gel electrophoresis, see section 21.4.

The following sections will describe these output formats in more detail.

In order to complete the analysis click **Finish** (see section 9.2 for information about the Save and Open options).

Restriction sites as annotation on the sequence

If you chose to add the restriction sites as annotation to the sequence, the result will be similar to the sequence shown in figure 21.44. See section 10.3 for more information about viewing annotations.

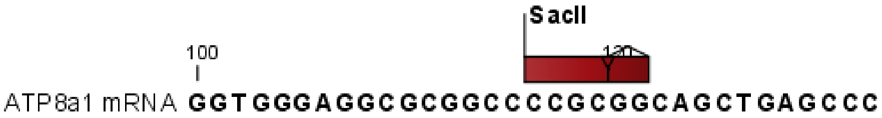
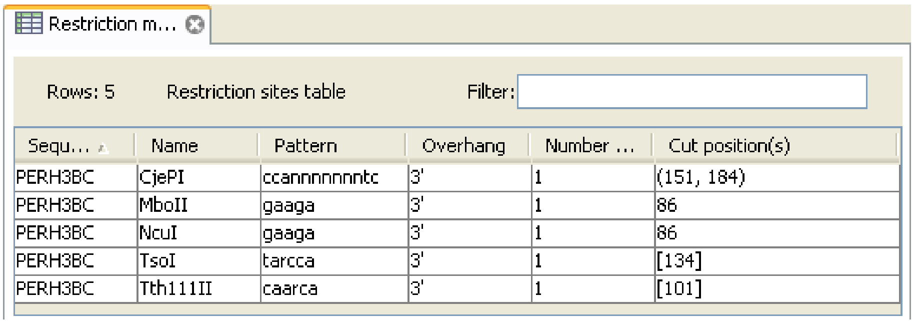


Figure 21.44: *The result of the restriction analysis shown as annotations.*

Table of restriction sites

The restriction map can be shown as a table of restriction sites (see figure 21.45).

Each row in the table represents a restriction enzyme. The following information is available for each enzyme:



Rows: 5 Restriction sites table Filter:

Sequ...	Name	Pattern	Overhang	Number ...	Cut position(s)
PERH3BC	CjePI	ccannnnnnntc	3'	1	(151, 184)
PERH3BC	MboII	gaaga	3'	1	86
PERH3BC	NcuI	gaaga	3'	1	86
PERH3BC	TsoI	tarcca	3'	1	[134]
PERH3BC	Tth111II	caarca	3'	1	[101]

Figure 21.45: *The result of the restriction analysis shown as annotations.*

- **Sequence**. The name of the sequence which is relevant if you have performed restriction map analysis on more than one sequence.
- **Name**. The name of the enzyme.
- **Pattern**. The recognition sequence of the enzyme.
- **Overhang**. The overhang produced by cutting with the enzyme (3', 5' or Blunt).
- **Number of cut sites**.
- **Cut position(s)**. The position of each cut.
  - **,** If the enzyme cuts more than once, the positions are separated by commas.
  - **[]** If the enzyme's recognition sequence is on the negative strand, the cut position is put in brackets (as the enzyme TsoI in figure 21.45 whose cut position is [134]).
  - **()** Some enzymes cut the sequence twice for each recognition site, and in this case the two cut positions are surrounded by parentheses.

Table of restriction fragments

The restriction map can be shown as a table of fragments produced by cutting the sequence with the enzymes:

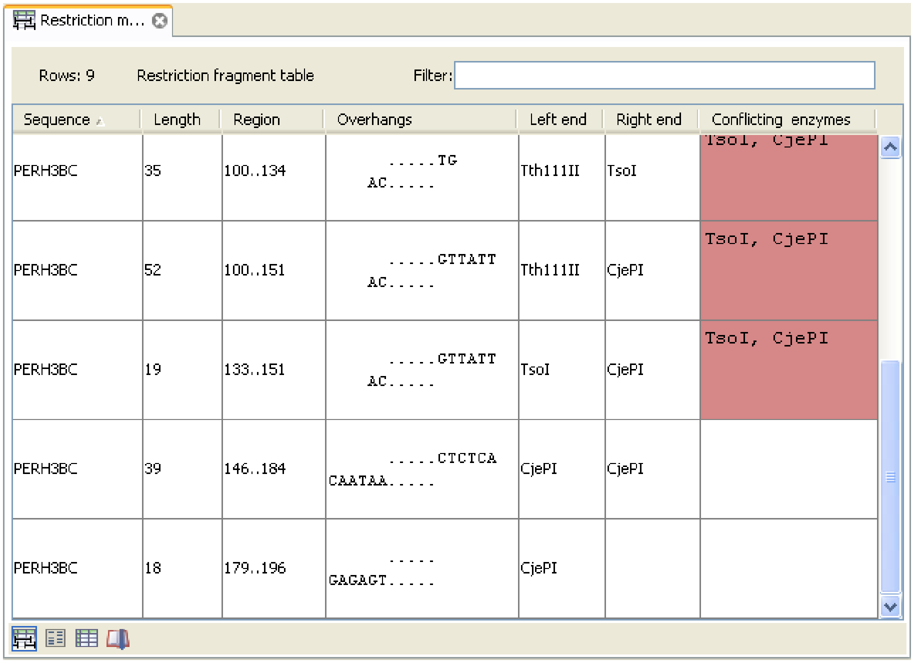
Click the Fragments button (▦) at the bottom of the view

The table is shown in see figure 21.46.

Each row in the table represents a fragment. If more than one enzyme cuts in the same region, or if an enzyme's recognition site is cut by another enzyme, there will be a fragment for each of the possible cut combinations [6]. The following information is available for each fragment.

- **Sequence**. The name of the sequence which is relevant if you have performed restriction map analysis on more than one sequence.
- **Length**. The length of the fragment. If there are overhangs of the fragment, these are included in the length (both 3' and 5' overhangs).
- **Region**. The fragment's region on the original sequence.

[6] Furthermore, if this is the case, you will see the names of the other enzymes in the **Conflicting Enzymes** column



Rows: 9 Restriction fragment table Filter:

Sequence	Length	Region	Overhangs	Left end	Right end	Conflicting enzymes
PERH3BC	35	100..134	.....TG AC.....	Tth111II	TsoI	TsoI, CjePI
PERH3BC	52	100..151	.....GTTATT AC.....	Tth111II	CjePI	TsoI, CjePI
PERH3BC	19	133..151	.....GTTATT AC.....	TsoI	CjePI	TsoI, CjePI
PERH3BC	39	146..184	.....CTCTCA CAATAA.....	CjePI	CjePI	
PERH3BC	18	179..196	..... GAGAGT.....	CjePI		

Figure 21.46: *The result of the restriction analysis shown as annotations.*

- **Overhangs**. If there is an overhang, this is displayed with an abbreviated version of the fragment and its overhangs. The two rows of dots (.) represent the two strands of the fragment and the overhang is visualized on each side of the dots with the residue(s) that make up the overhang. If there are only the two rows of dots, it means that there is no overhang.
- **Left end**. The enzyme that cuts the fragment to the left (5' end).
- **Right end**. The enzyme that cuts the fragment to the right (3' end).
- **Conflicting enzymes**. If more than one enzyme cuts at the same position, or if an enzyme's recognition site is cut by another enzyme, a fragment is displayed for each possible combination of cuts. At the same time, this column will display the enzymes that are in conflict. If there are conflicting enzymes, they will be colored red to alert the user. If the same experiment were performed in the lab, conflicting enzymes could lead to wrong results. For this reason, this functionality is useful to simulate digestions with complex combinations of restriction enzymes.

If views of both the fragment table and the sequence are open, clicking in the fragment table will select the corresponding region on the sequence.

Gel

The restriction map can also be shown as a gel. This is described in section 21.4.1.

21.4 Gel electrophoresis

CLC Genomics Workbench enables the user to simulate the separation of nucleotide sequences on a gel. This feature is useful when e.g. designing an experiment which will allow the differentiation of a successful and an unsuccessful cloning experiment on the basis of a restriction map.

There are two main ways to simulate gel separation of nucleotide sequences:

- One or more sequences can be digested with restriction enzymes and the resulting fragments can be separated on a gel.
- A number of existing sequences can be separated on a gel.

There are several ways to apply these functionalities as described below.

21.4.1 Separate fragments of sequences on gel

This section explains how to simulate a gel electrophoresis of one or more sequences which are digested with restriction enzymes. There are two ways to do this:

- When performing the **Restriction Site Analysis** from the **Toolbox**, you can choose to create a restriction map which can be shown as a gel. This is explained in section 21.3.2.
- From all the graphical views of sequences, you can right-click the name of the sequence and choose: **Digest Sequence with Selected Enzymes and Run on Gel (▤)**. The views where this option is available are listed below:
  - Circular view (see section 10.2).
  - Ordinary sequence view (see section 10.1).
  - Graphical view of sequence lists (see section 10.7).
  - Cloning editor (see section 21.1).
  - Primer designer (see section 17.3).

  Furthermore, you can also right-click an empty part of the view of the graphical view of sequence lists and the cloning editor and choose **Digest All Sequences with Selected Enzymes and Run on Gel**.
  Note! When using the right-click options, the sequence will be digested with the enzymes that are selected in the **Side Panel**. This is explained in section 10.1.2.

The view of the gel is explained in section 21.4.3

21.4.2 Separate sequences on gel

To separate sequences without restriction enzyme digestion, first create a sequence list of the sequences in question (see section 10.7). Then click the **Gel** button (▤) at the bottom of the view of the sequence list.

For more information about the view of the gel, see the next section.

21.4.3 Gel view

In figure 21.48 you can see a simulation of a gel with its **Side Panel** to the right. This view will be explained in this section.

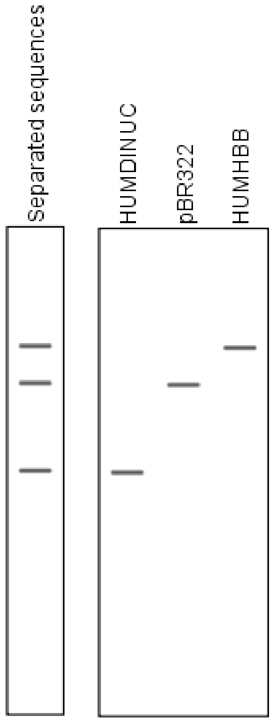
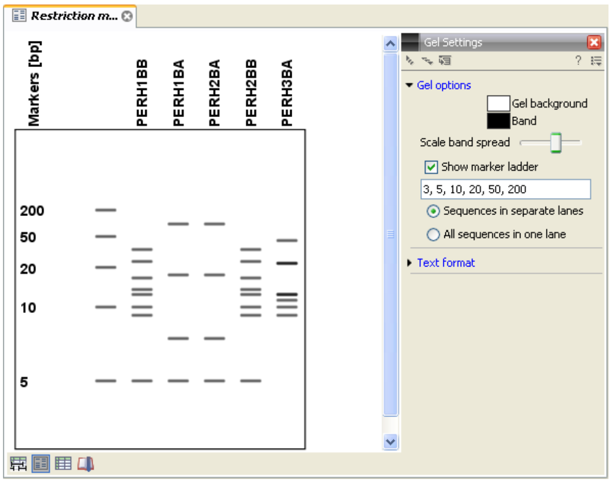


Figure 21.47: *A sequence list shown as a gel.*



Gel background
Band
Scale band spread
☑ Show marker ladder
3, 5, 10, 20, 50, 200
● Sequences in separate lanes
○ All sequences in one lane

Figure 21.48: *Five lanes showing fragments of five sequences cut with restriction enzymes.*

Information on bands / fragments

You can get information about the individual bands by hovering the mouse cursor on the band of interest. This will display a tool tip with the following information:

- Fragment length
- Fragment region on the original sequence
- Enzymes cutting at the left and right ends, respectively

For gels comparing whole sequences, you will see the sequence name and the length of the sequence.

Note! You have to be in **Selection** ( ) or **Pan** ( ) mode in order to get this information.

It can be useful to add markers to the gel which enables you to compare the sizes of the bands. This is done by clicking **Show marker ladder** in the **Side Panel**.

Markers can be entered into the text field, separated by commas.

Modifying the layout

The background of the lane and the colors of the bands can be changed in the **Side Panel**. Click the colored box to display a dialog for picking a color. The slider **Scale band spread** can be used to adjust the effective time of separation on the gel, i.e. how much the bands will be spread over the lane. In a real electrophoresis experiment this property will be determined by several factors including time of separation, voltage and gel density.

You can also choose how many lanes should be displayed:

- **Sequences in separate lanes.** This simulates that a gel is run for each sequence.
- **All sequences in one lane**. This simulates that one gel is run for all sequences.

You can also modify the layout of the view by zooming in or out. Click **Zoom in** ( ) or **Zoom out** ( ) in the Toolbar and click the view.

Finally, you can modify the format of the text heading each lane in the **Text format** preferences in the **Side Panel**.

21.5 Restriction enzyme lists

CLC Genomics Workbench includes all the restriction enzymes available in the **REBASE** database[7]. However, when performing restriction site analyses, it is often an advantage to use a customized list of enzymes. In this case, the user can create special lists containing e.g. all enzymes available in the laboratory freezer, all enzymes used to create a given restriction map or all enzymes that are available form the preferred vendor.

In the example data (see section 1.6.2) under Nucleotide->Restriction analysis, there are two enzyme lists: one with the 50 most popular enzymes, and another with all enzymes that are included in the *CLC Genomics Workbench*.

This section describes how you can create an enzyme list, and how you can modify it.

21.5.1 Create enzyme list

CLC Genomics Workbench uses enzymes from the **REBASE** restriction enzyme database at `http://rebase.neb.com`[8].

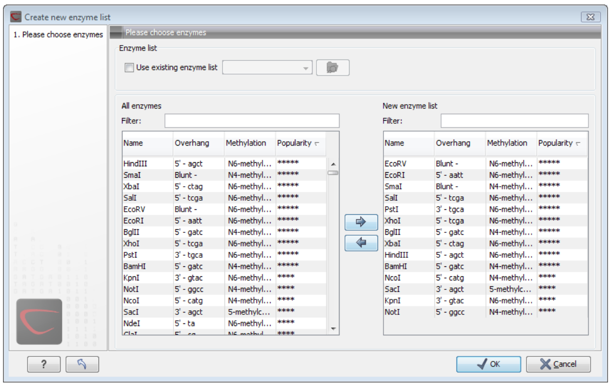
To create an enzyme list of a subset of these enzymes:

[7] You can customize the enzyme database for your installation, see section F

[8] You can customize the enzyme database for your installation, see section F

File | **New** | **Enzyme list** (▦)

This opens the dialog shown in figure 21.49



All enzymes

Name	Overhang	Methylation	Popularity
HindIII	5' - agct	N6-methyl...	*****
SmaI	Blunt -	N4-methyl...	*****
XbaI	5' - ctag	N6-methyl...	*****
SalI	5' - tcga	N6-methyl...	*****
EcoRV	Blunt -	N6-methyl...	*****
EcoRI	5' - aatt	N6-methyl...	*****
BglII	5' - gatc	N4-methyl...	*****
XhoI	5' - tcga	N6-methyl...	*****
PstI	3' - tgca	N6-methyl...	*****
BamHI	5' - gatc	N4-methyl...	*****
KpnI	3' - gtac	N6-methyl...	****
NotI	5' - ggcc	N4-methyl...	****
NcoI	5' - catg	N4-methyl...	****
SacI	3' - agct	5-methylc...	****
NdeI	5' - ta	N6-methyl...	****
ClaI	5' - cg	N6-methyl...	****

New enzyme list

Name	Overhang	Methylation	Popularity
EcoRV	Blunt -	N6-methyl...	*****
EcoRI	5' - aatt	N6-methyl...	*****
SmaI	Blunt -	N4-methyl...	*****
SalI	5' - tcga	N6-methyl...	*****
PstI	3' - tgca	N6-methyl...	*****
XhoI	5' - tcga	N6-methyl...	*****
BglII	5' - gatc	N4-methyl...	*****
XbaI	5' - ctag	N6-methyl...	*****
HindIII	5' - agct	N6-methyl...	*****
BamHI	5' - gatc	N4-methyl...	*****
NcoI	5' - catg	N4-methyl...	****
SacI	3' - agct	5-methylc...	****
KpnI	3' - gtac	N6-methyl...	****
NotI	5' - ggcc	N4-methyl...	****

Figure 21.49: *Choosing enzymes for the new enzyme list.*

At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. See section 21.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you see all the enzymes that are in the list select above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [9].
- To the **right**, there is a list of the enzymes that will be used.

Select enzymes in the left side panel and add them to the right panel by double-clicking or clicking the **Add** button (➡). If you e.g. wish to use EcoRV and BamHI, select these two enzymes and add them to the right side panel.

If you wish to use all the enzymes in the list:

Click in the panel to the left | **press Ctrl + A (⌘ + A on Mac)** | **Add (➡)**

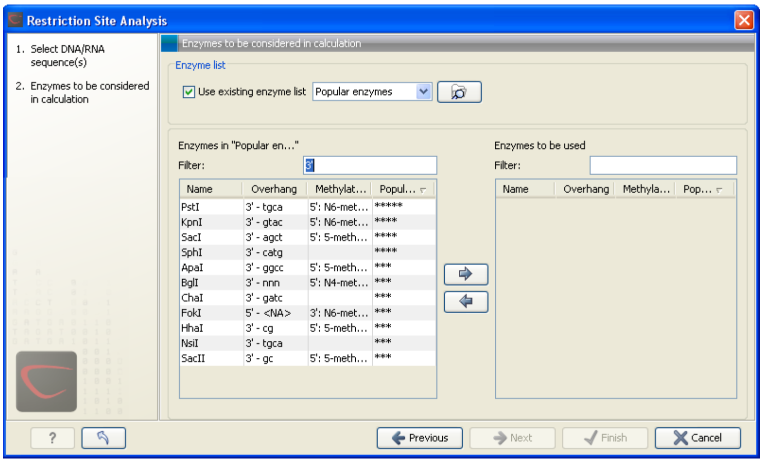
The enzymes can be sorted by clicking the column headings, i.e. Name, Overhang, Methylation or Popularity. This is particularly useful if you wish to use enzymes which produce e.g. a 3' overhang. In this case, you can sort the list by clicking the Overhang column heading, and all the enzymes producing 3' overhangs will be listed together for easy selection.

When looking for a specific enzyme, it is easier to use the Filter. If you wish to find e.g. HindIII sites, simply type HindIII into the filter, and the list of enzymes will shrink automatically to only include the HindIII enzyme. This can also be used to only show enzymes producing e.g. a 3' overhang as shown in figure 21.50.

If you need more detailed information and filtering of the enzymes, either place your mouse cursor on an enzyme for one second to display additional information (see figure 21.51), or use the view of enzyme lists (see 21.5).

Click **Finish** to open the enzyme list.

[9] The *CLC Genomics Workbench* comes with a standard set of enzymes based on http://www.rebase.neb.com. You can customize the enzyme database for your installation, see section F



Restriction Site Analysis
1. Select DNA/RNA sequence(s)
2. Enzymes to be considered in calculation
Enzymes to be considered in calculation
Enzyme list
?
Previous
Next
Finish
Cancel

☑ Use existing enzyme list: Popular enzymes

Enzymes in "Popular en..."

Filter: 3'

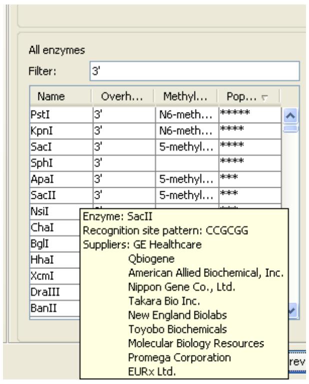
Name	Overhang	Methylat...	Popul...
PstI	3' - tgca	5': N6-met...	*****
KpnI	3' - gtac	5': N6-met...	****
SacI	3' - agct	5': 5-meth...	****
SphI	3' - catg		****
ApaI	3' - ggcc	5': 5-meth...	***
BglI	3' - nnn	5': N4-met...	***
ChaI	3' - gatc		***
FokI	5' - <NA>	3': N6-met...	***
HhaI	3' - cg	5': 5-meth...	***
NsiI	3' - tgca		***
SacII	3' - gc	5': 5-meth...	***

Enzymes to be used

Filter:

Name	Overhang	Methyla...	Pop...

Figure 21.50: *Selecting enzymes.*



All enzymes

Filter: 3'

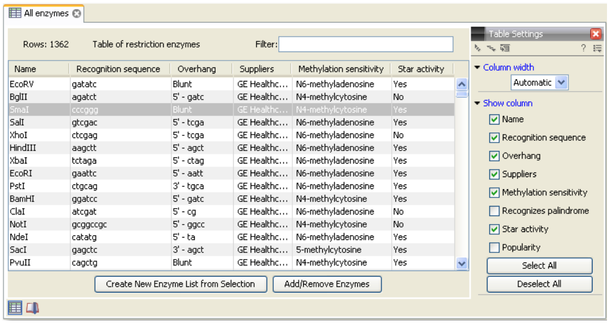
Name	Overh...	Methyl...	Pop...
PstI	3'	N6-meth...	*****
KpnI	3'	N6-meth...	****
SacI	3'	5-methyl...	****
SphI	3'		****
ApaI	3'	5-methyl...	***
SacII	3'	5-methyl...	***
NsiI			
ChaI			
BglI			
HhaI			
XcmI			
DraIII			
BanII			

Enzyme: SacII
Recognition site pattern: CCGCGG
Suppliers: GE Healthcare
Qbiogene
American Allied Biochemical, Inc.
Nippon Gene Co., Ltd.
Takara Bio Inc.
New England Biolabs
Toyobo Biochemicals
Molecular Biology Resources
Promega Corporation
EURx Ltd.

Figure 21.51: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

21.5.2 View and modify enzyme list

An enzyme list is shown in figure 21.52. The list can be sorted by clicking the columns,



Rows: 1362 Table of restriction enzymes Filter:

Name	Recognition sequence	Overhang	Suppliers	Methylation sensitivity	Star activity
EcoRV	gatatc	Blunt	GE Healthc...	N6-methyladenosine	Yes
BglII	agatct	5' - gatc	GE Healthc...	N4-methylcytosine	No
SmaI	cccggg	Blunt	GE Healthc...	N4-methylcytosine	Yes
SalI	gtcgac	5' - tcga	GE Healthc...	N6-methyladenosine	Yes
XhoI	ctcgag	5' - tcga	GE Healthc...	N6-methyladenosine	No
HindIII	aagctt	5' - agct	GE Healthc...	N6-methyladenosine	Yes
XbaI	tctaga	5' - ctag	GE Healthc...	N6-methyladenosine	Yes
EcoRI	gaattc	5' - aatt	GE Healthc...	N6-methyladenosine	Yes
PstI	ctgcag	3' - tgca	GE Healthc...	N6-methyladenosine	Yes
BamHI	ggatcc	5' - gatc	GE Healthc...	N4-methylcytosine	Yes
ClaI	atcgat	5' - cg	GE Healthc...	N6-methyladenosine	No
NotI	gcggccgc	5' - ggcc	GE Healthc...	N4-methylcytosine	No
NdeI	catatg	5' - ta	GE Healthc...	N6-methyladenosine	Yes
SacI	gagctc	3' - agct	GE Healthc...	5-methylcytosine	Yes
PvuII	cagctg	Blunt	GE Healthc...	N4-methylcytosine	Yes

Column width: Automatic

Show column
- ☑ Name
- ☑ Recognition sequence
- ☑ Overhang
- ☑ Suppliers
- ☑ Methylation sensitivity
- □ Recognizes palindrome
- ☑ Star activity
- □ Popularity

Figure 21.52: *An enzyme list.*

and you can use the filter at the top right corner to search for specific enzymes, recognition sequences etc.

If you wish to remove or add enzymes, click the **Add/Remove Enzymes** button at the bottom of the view. This will present the same dialog as shown in figure 21.49 with the enzyme list shown to the right.

If you wish to extract a subset of an enzyme list:

open the list | **select the relevant enzymes** | **right-click** | **Create New Enzyme List from Selection** ( )

If you combined this method with the filter located at the top of the view, you can extract a very specific set of enzymes. E.g. if you wish to create a list of enzymes sold by a particular distributor, type the name of the distributor into the filter, and select and create a new enzyme list from the selection.

Chapter 22

Sequence alignment

Contents

CLC Genomics Workbench can align nucleotides and proteins using a *progressive alignment* algorithm (see section 22.6 or read the White paper on alignments in the **Science** section of `http://www.clcbio.com`).

This chapter describes how to use the program to align sequences. The chapter also describes alignment algorithms in more general terms.

22.1 Create an alignment

Alignments can be created from sequences, sequence lists (see section 10.7), existing alignments and from any combination of the three.

To create an alignment in *CLC Genomics Workbench*:

> **select sequences to align | Toolbox in the Menu Bar | Alignments and Trees ( ) | Create Alignment ( )**

or **select sequences to align | right-click any selected sequence | Toolbox | Alignments and Trees ( ) | Create Alignment ( )**

This opens the dialog shown in figure 22.1.

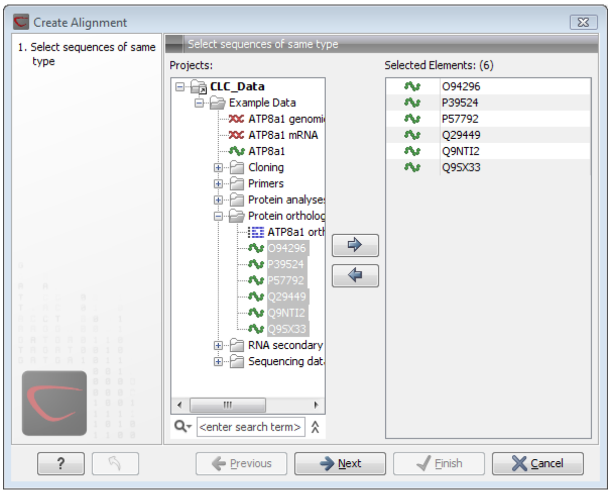
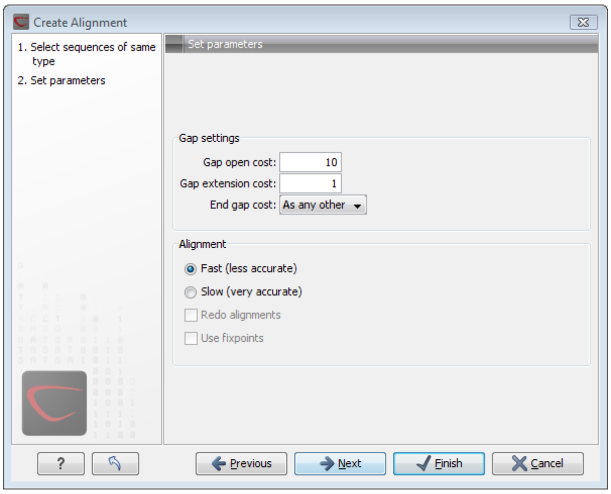


Figure 22.1: *Creating an alignment.*

If you have selected some elements before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences, sequence lists or alignments from the selected elements. Click **Next** to adjust alignment algorithm parameters. Clicking **Next** opens the dialog shown in figure 22.2.



Gap open cost: 10

Gap extension cost: 1

End gap cost: As any other

● Fast (less accurate)

○ Slow (very accurate)

□ Redo alignments

□ Use fixpoints

Figure 22.2: *Adjusting alignment algorithm parameters.*

22.1.1 Gap costs

The alignment algorithm has three parameters concerning gap costs: Gap open cost, Gap extension cost and End gap cost. The precision of these parameters is to one place of decimal.

- **Gap open cost**. The price for introducing gaps in an alignment.
- **Gap extension cost**. The price for every extension past the initial gap.

If you expect a lot of small gaps in your alignment, the Gap open cost should equal the Gap extension cost. On the other hand, if you expect few but large gaps, the Gap open cost should be set significantly higher than the Gap extension cost.

However, for most alignments it is a good idea to make the Gap open cost quite a bit higher than the Gap extension cost. The default values are 10.0 and 1.0 for the two parameters, respectively.

- **End gap cost**. The price of gaps at the beginning or the end of the alignment. One of the advantages of the *CLC Genomics Workbench* alignment method is that it provides flexibility in the treatment of gaps at the ends of the sequences. There are three possibilities:
  - **Free end gaps**. Any number of gaps can be inserted in the ends of the sequences without any cost.
  - **Cheap end gaps**. All end gaps are treated as gap extensions and any gaps past 10 are free.
  - **End gaps as any other**. Gaps at the ends of sequences are treated like gaps in any other place in the sequences.

When aligning a long sequence with a short partial sequence, it is ideal to use free end gaps, since this will be the best approximation to the situation. The many gaps inserted at the ends are not due to evolutionary events, but rather to partial data.

Many homologous proteins have quite different ends, often with large insertions or deletions. This confuses alignment algorithms, but using the **Cheap end gaps** option, large gaps will generally be tolerated at the sequence ends, improving the overall alignment. This is the default setting of the algorithm.

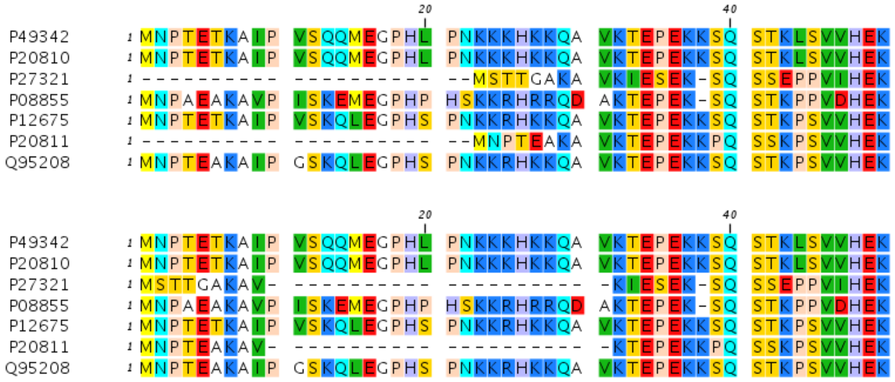
Finally, treating end gaps like any other gaps is the best option when you know that there are no biologically distinct effects at the ends of the sequences.

Figures 22.3 and 22.4 illustrate the differences between the different gap scores at the sequence ends.

22.1.2 Fast or accurate alignment algorithm

CLC Genomics Workbench has two algorithms for calculating alignments:

- **Fast (less accurate).** This allows for use of an optimized alignment algorithm which is very fast. The fast option is particularly useful for data sets with very long sequences.
- **Slow (very accurate).** This is the recommended choice unless you find the processing time too long.



```
                                  20                    40
P49342  1 MNPTETKAIP VSQQMEGPHL PNKKKHKKQA VKTEPEKKSQ STKLSVVHEK
P20810  1 MNPTETKAIP VSQQMEGPHL PNKKKHKKQA VKTEPEKKSQ STKLSVVHEK
P27321  1 ---------- ---------- --MSTTGAKA VKIESEK-SQ SSEPPVIHEK
P08855  1 MNPAEAKAVP ISKEMEGPHP HSKKRHRRQD AKTEPEK-SQ STKPPVDHEK
P12675  1 MNPTETKAIP VSKQLEGPHS PNKKRHKKQA VKTEPEKKSQ STKPSVVHEK
P20811  1 ---------- ---------- --MNPTEAKA VKTEPEKKPQ SSKPSVVHEK
Q95208  1 MNPTEAKAIP GSKQLEGPHS PNKKRHKKQA VKTEPEKKSQ STKPSVVHEK

                                  20                    40
P49342  1 MNPTETKAIP VSQQMEGPHL PNKKKHKKQA VKTEPEKKSQ STKLSVVHEK
P20810  1 MNPTETKAIP VSQQMEGPHL PNKKKHKKQA VKTEPEKKSQ STKLSVVHEK
P27321  1 MSTTGAKAV- ---------- ---------- -KIESEK-SQ SSEPPVIHEK
P08855  1 MNPAEAKAVP ISKEMEGPHP HSKKRHRRQD AKTEPEK-SQ STKPPVDHEK
P12675  1 MNPTETKAIP VSKQLEGPHS PNKKRHKKQA VKTEPEKKSQ STKPSVVHEK
P20811  1 MNPTEAKAV- ---------- ---------- -KTEPEKKPQ SSKPSVVHEK
Q95208  1 MNPTEAKAIP GSKQLEGPHS PNKKRHKKQA VKTEPEKKSQ STKPSVVHEK
```

Figure 22.3: *The first 50 positions of two different alignments of seven calpastatin sequences. The top alignment is made with cheap end gaps, while the bottom alignment is made with end gaps having the same price as any other gaps. In this case it seems that the latter scoring scheme gives the best result.*

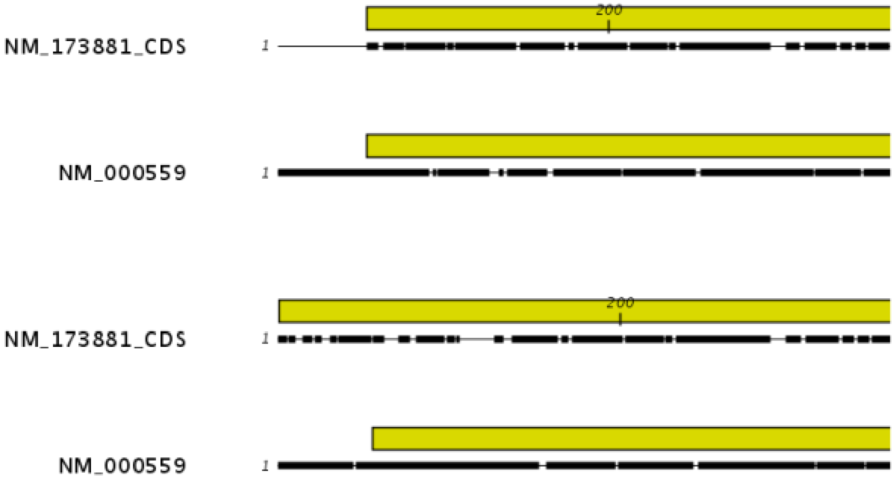


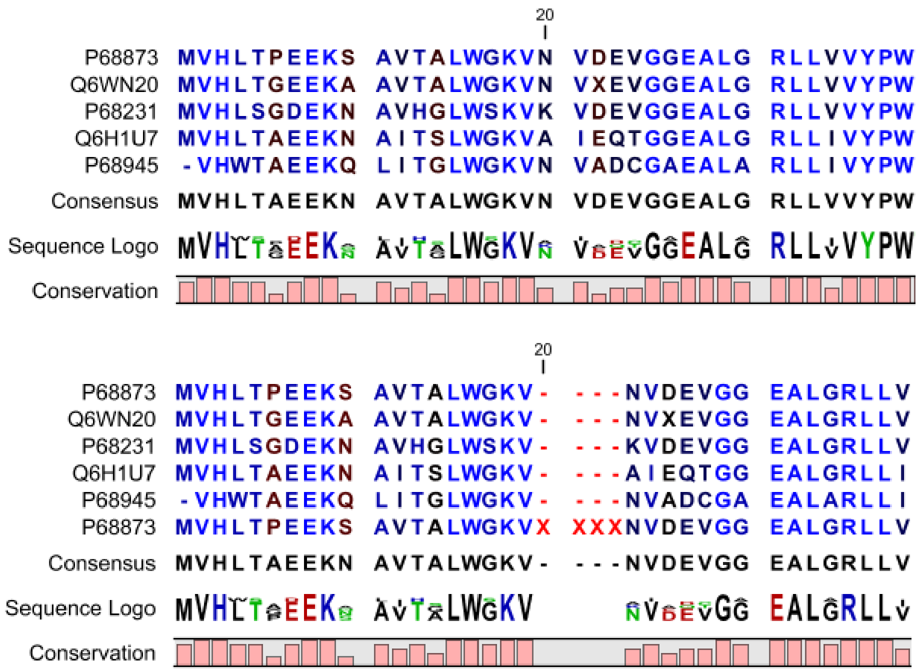
Figure 22.4: *The alignment of the coding sequence of bovine myoglobin with the full mRNA of human gamma globin. The top alignment is made with free end gaps, while the bottom alignment is made with end gaps treated as any other. The yellow annotation is the coding sequence in both sequences. It is evident that free end gaps are ideal in this situation as the start codons are aligned correctly in the top alignment. Treating end gaps as any other gaps in the case of aligning distant homologs where one sequence is partial leads to a spreading out of the short sequence as in the bottom alignment.*

Both algorithms use progressive alignment. The faster algorithm builds the initial tree by doing more approximate pairwise alignments than the slower option.

22.1.3 Aligning alignments

If you have selected an existing alignment in the first step (22.1), you have to decide how this alignment should be treated.

- **Redo alignment.** The original alignment will be realigned if this checkbox is checked. Otherwise, the original alignment is kept in its original form except for possible extra equally sized gaps in all sequences of the original alignment. This is visualized in figure 22.5.



```
                                     20
                                     |
P68873         MVHLTPEEKS AVTALWGKVN VDEVGGEALG RLLVVYPW
Q6WN20         MVHLTGEEKA AVTALWGKVN VXEVGGEALG RLLVVYPW
P68231         MVHLSGDEKN AVHGLWSKVK VDEVGGEALG RLLVVYPW
Q6H1U7         MVHLTAEEKN AITSLWGKVA IEQTGGEALG RLLIVYPW
P68945         -VHWTAEEKQ LITGLWGKVN VADCGAEALA RLLIVYPW
Consensus      MVHLTAEEKN AVTALWGKVN VDEVGGEALG RLLVVYPW
Sequence Logo
Conservation

                                     20
                                     |
P68873         MVHLTPEEKS AVTALWGKV- ---NVDEVGG EALGRLLV
Q6WN20         MVHLTGEEKA AVTALWGKV- ---NVXEVGG EALGRLLV
P68231         MVHLSGDEKN AVHGLWSKV- ---KVDEVGG EALGRLLV
Q6H1U7         MVHLTAEEKN AITSLWGKV- ---AIEQTGG EALGRLLI
P68945         -VHWTAEEKQ LITGLWGKV- ---NVADCGA EALARLLI
P68873         MVHLTPEEKS AVTALWGKVX XXXNVDEVGG EALGRLLV
Consensus      MVHLTAEEKN AVTALWGKV- ---NVDEVGG EALGRLLV
Sequence Logo
Conservation
```

Figure 22.5: *The top figures shows the original alignment. In the bottom panel a single sequence with four inserted X's are aligned to the original alignment. This introduces gaps in all sequences of the original alignment. All other positions in the original alignment are fixed.*

This feature is useful if you wish to add extra sequences to an existing alignment, in which case you just select the alignment and the extra sequences and choose not to redo the alignment.

It is also useful if you have created an alignment where the gaps are not placed correctly. In this case, you can realign the alignment with different gap cost parameters.

22.1.4 Fixpoints

With fixpoints, you can get full control over the alignment algorithm. The fixpoints are points on the sequences that are forced to align to each other.

Fixpoints are added to sequences or alignments before clicking "Create alignment". To add a fixpoint, open the sequence or alignment and:

> **Select the region you want to use as a fixpoint | right-click the selection | Set alignment fixpoint here**

This will add an annotation labeled "Fixpoint" to the sequence (see figure 22.6). Use this procedure to add fixpoints to the other sequence(s) that should be forced to align to each other.

When you click "Create alignment" and go to **Step 2**, check **Use fixpoints** in order to force the alignment algorithm to align the fixpoints in the selected sequences to each other.

In figure 22.7 the result of an alignment using fixpoints is illustrated.

You can add multiple fixpoints, e.g. adding two fixpoints to the sequences that are aligned will force their first fixpoints to be aligned to each other, and their second fixpoints will also be

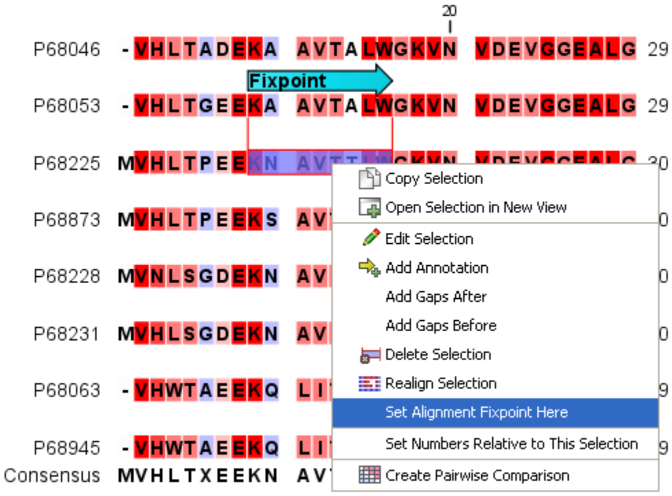


Figure 22.6: *Adding a fixpoint to a sequence in an existing alignment. At the top you can see a fixpoint that has already been added.*

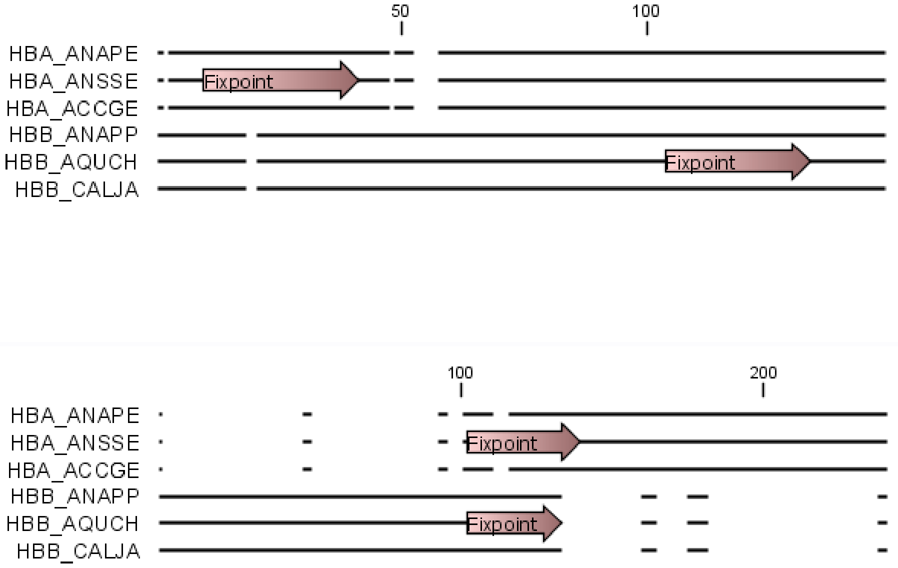


Figure 22.7: *Realigning using fixpoints. In the top view, fixpoints have been added to two of the sequences. In the view below, the alignment has been realigned using the fixpoints. The three top sequences are very similar, and therefore they follow the one sequence (number two from the top) that has a fixpoint.*

aligned to each other.

Advanced use of fixpoints

Fixpoints with the same names will be aligned to each other, which gives the opportunity for great control over the alignment process. It is only necessary to change any fixpoint names in very special cases.

One example would be three sequences A, B and C where sequences A and B has one copy of a domain while sequence C has two copies of the domain. You can now force sequence A to align to the first copy and sequence B to align to the second copy of the domains in sequence C. This is done by inserting fixpoints in sequence C for each domain, and naming them 'fp1' and 'fp2' (for example). Now, you can insert a fixpoint in each of sequences A and B, naming them 'fp1' and 'fp2', respectively. Now, when aligning the three sequences using fixpoints, sequence A will align to the first copy of the domain in sequence C, while sequence B would align to the second copy of the domain in sequence C.

You can name fixpoints by:

> **right-click the Fixpoint annotation | Edit Annotation ( ) | type the name in the 'Name' field**

22.2 View alignments

Since an alignment is a display of several sequences arranged in rows, the basic options for viewing alignments are the same as for viewing sequences. Therefore we refer to section 10.1 for an explanation of these basic options.

However, there are a number of alignment-specific view options in the **Alignment info** and the **Nucleotide info** in the **Side Panel** to the right of the view. Below is more information on these view options.

Under **Translation** in the **Nucleotide info**, there is an extra checkbox: **Relative to top sequence**. Checking this box will make the reading frames for the translation align with the top sequence so that you can compare the effect of nucleotide differences on the protein level.

The options in the **Alignment info** relate to each column in the alignment:

- **Consensus.** Shows a consensus sequence at the bottom of the alignment. The consensus sequence is based on every single position in the alignment and reflects an artificial sequence which resembles the sequence information of the alignment, but only as one single sequence. If all sequences of the alignment is 100% identical the consensus sequence will be identical to all sequences found in the alignment. If the sequences of the alignment differ the consensus sequence will reflect the most common sequences in the alignment. Parameters for adjusting the consensus sequences are described below.
  - **Limit.** This option determines how conserved the sequences must be in order to agree on a consensus. Here you can also choose **IUPAC** which will display the ambiguity code when there are differences between the sequences. E.g. an alignment with **A** and a **G** at the same position will display an **R** in the consensus line if the **IUPAC** option is selected. (The IUPAC codes can be found in section I and H.)
  - **No gaps.** Checking this option will not show gaps in the consensus.
  - **Ambiguous symbol.** Select how ambiguities should be displayed in the consensus line (as **N**, **?**, *****, **.** or **-**). This option has now effect if **IUPAC** is selected in the **Limit** list above.

  The **Consensus Sequence** can be opened in a new view, simply by right-clicking the **Consensus Sequence** and click **Open Consensus in New View**.

- **Conservation.** Displays the level of conservation at each position in the alignment. The conservation shows the conservation of all sequence positions. The height of the bar, or the gradient of the color reflect how conserved that particular position is in the alignment. If one position is 100% conserved the bar will be shown in full height, and it is colored in the color specified at the right side of the gradient slider.

  - **Foreground color.** Colors the letters using a gradient, where the right side color is used for highly conserved positions and the left side color is used for positions that are less conserved.
  - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
  - **Graph.** Displays the conservation level as a graph at the bottom of the alignment. The bar (default view) show the conservation of all sequence positions. The height of the graph reflects how conserved that particular position is in the alignment. If one position is 100% conserved the graph will be shown in full height. Learn how to export the data behind the graph in section 7.4.
    - **Height.** Specifies the height of the graph.
    - **Type.** The type of the graph.
      - **Line plot.** Displays the graph as a line plot.
      - **Bar plot.** Displays the graph as a bar plot.
      - **Colors.** Displays the graph as a color bar using a gradient like the foreground and background colors.
    - **Color box.** Specifies the color of the graph for line and bar plots, and specifies a gradient for colors.

- **Gap fraction.** Which fraction of the sequences in the alignment that have gaps. The gap fraction is only relevant if there are gaps in the alignment.
  - **Foreground color.** Colors the letter using a gradient, where the left side color is used if there are relatively few gaps, and the right side color is used if there are relatively many gaps.
  - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
  - **Graph.** Displays the gap fraction as a graph at the bottom of the alignment (Learn how to export the data behind the graph in section 7.4).
    - **Height.** Specifies the height of the graph.
    - **Type.** The type of the graph.
      - **Line plot.** Displays the graph as a line plot.
      - **Bar plot.** Displays the graph as a line plot.
      - **Colors.** Displays the graph as a color bar using a gradient like the foreground and background colors.
    - **Color box.** Specifies the color of the graph for line and bar plots, and specifies a gradient for colors.

- **Color different residues.** Indicates differences in aligned residues.
  - **Foreground color.** Colors the letter.
  - **Background color.** Sets a background color of the residues.

- **Sequence logo.** A sequence logo displays the frequencies of residues at each position in an alignment. This is presented as the relative heights of letters, along with the degree of sequence conservation as the total height of a stack of letters, measured in bits of information. The vertical scale is in bits, with a maximum of 2 bits for nucleotides and approximately 4.32 bits for amino acid residues. See section 22.2.1 for more details.

- **Foreground color.** Color the residues using a gradient according to the information content of the alignment column. Low values indicate columns with high variability whereas high values indicate columns with similar residues.
- **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
- **Logo.** Displays sequence logo at the bottom of the alignment.
  - **Height.** Specifies the height of the sequence logo graph.
  - **Color.** The sequence logo can be displayed in black or Rasmol colors. For protein alignments, a polarity color scheme is also available, where hydrophobic residues are shown in black color, hydrophilic residues as green, acidic residues as red and basic residues as blue.

22.2.1 Bioinformatics explained: Sequence logo

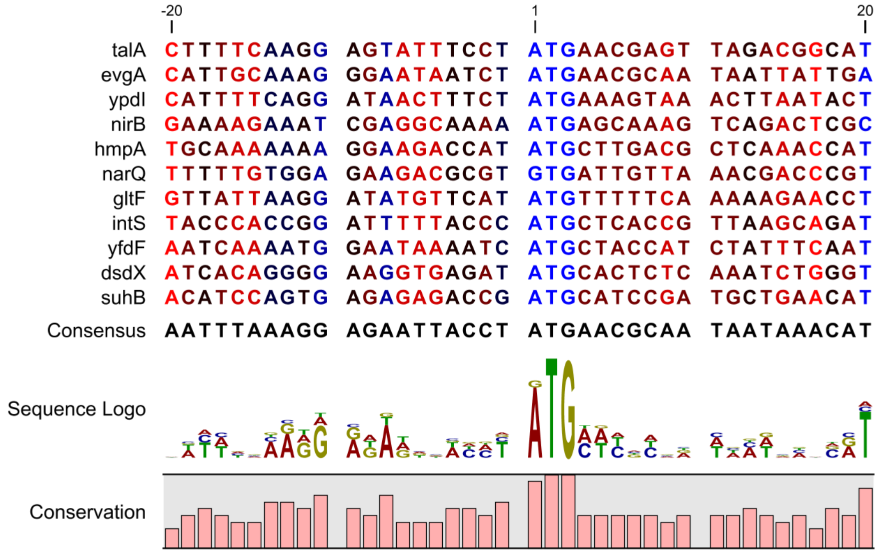
In the search for homologous sequences, researchers are often interested in conserved sites/residues or positions in a sequence which tend to differ a lot. Most researches use alignments (see Bioinformatics explained*: multiple alignments*) for visualization of homology on a given set of either DNA or protein sequences. In proteins, active sites in a given protein family are often highly conserved. Thus, in an alignment these positions (which are not necessarily located in proximity) are fully or nearly fully conserved. On the other hand, antigen binding sites in the F$_{ab}$ unit of immunoglobulins tend to differ quite a lot, whereas the rest of the protein remains relatively unchanged.

In DNA, promoter sites or other DNA binding sites are highly conserved (see figure 22.8). This is also the case for repressor sites as seen for the Cro repressor of bacteriophage λ.

When aligning such sequences, regardless of whether they are highly variable or highly conserved at specific sites, it is very difficult to generate a consensus sequence which covers the actual variability of a given position. In order to better understand the information content or significance of certain positions, a sequence logo can be used. The sequence logo displays the information content of all positions in an alignment as residues or nucleotides stacked on top of each other (see figure 22.8). The sequence logo provides a far more detailed view of the entire alignment than a simple consensus sequence. Sequence logos can aid to identify protein binding sites on DNA sequences and can also aid to identify conserved residues in aligned domains of protein sequences and a wide range of other applications.

Each position of the alignment and consequently the sequence logo shows the sequence information in a computed score based on Shannon entropy [Schneider and Stephens, 1990]. The height of the individual letters represent the sequence information content in that particular position of the alignment.

A sequence logo is a much better visualization tool than a simple consensus sequence. An example hereof is an alignment where in one position a particular residue is found in 70% of the sequences. If a consensus sequence is used, it typically only displays the single residue with 70% coverage. In figure 22.8 an un-gapped alignment of 11 *E. coli* start codons including flanking regions are shown. In this example, a consensus sequence would only display ATG as the start codon in position 1, but when looking at the sequence logo it is seen that a GTG is also allowed as a start codon.



```
talA       CTTTTCAAGG  AGTATTTCCT  ATGAACGAGT  TAGACGGCAT
evgA       CATTGCAAAG  GGAATAATCT  ATGAACGCAA  TAATTATTGA
ypdI       CATTTTCAGG  ATAACTTTCT  ATGAAAGTAA  ACTTAATACT
nirB       GAAAAGAAAT  CGAGGCAAAA  ATGAGCAAAG  TCAGACTCGC
hmpA       TGCAAAAAAA  GGAAGACCAT  ATGCTTGACG  CTCAAACCAT
narQ       TTTTTGTGGA  GAAGACGCGT  GTGATTGTTA  AACGACCCGT
gltF       GTTATTAAGG  ATATGTTCAT  ATGTTTTTCA  AAAAGAACCT
intS       TACCCACCGG  ATTTTTACCC  ATGCTCACCG  TTAAGCAGAT
yfdF       AATCAAAATG  GAATAAAATC  ATGCTACCAT  CTATTTCAAT
dsdX       ATCACAGGGG  AAGGTGAGAT  ATGCACTCTC  AAATCTGGGT
suhB       ACATCCAGTG  AGAGAGACCG  ATGCATCCGA  TGCTGAACAT

Consensus  AATTTAAAGG  AGAATTACCT  ATGAACGCAA  TAATAAACAT
```

Figure 22.8: *Ungapped sequence alignment of eleven* E. coli *sequences defining a start codon. The start codons start at position 1. Below the alignment is shown the corresponding sequence logo. As seen, a GTG start codon and the usual ATG start codons are present in the alignment. This can also be visualized in the logo at position 1.*

Calculation of sequence logos

A comprehensive walk-through of the calculation of the information content in sequence logos is beyond the scope of this document but can be found in the original paper by [Schneider and Stephens, 1990]. Nevertheless, the conservation of every position is defined as R_{seq} which is the difference between the maximal entropy (S_{max}) and the observed entropy for the residue distribution (S_{obs}),

$$R_{seq} = S_{max} - S_{obs} = \log_2 N - \left(- \sum_{n=1}^{N} p_n \log_2 p_n \right)$$

p_n is the observed frequency of a amino acid residue or nucleotide of symbol n at a particular position and N is the number of distinct symbols for the sequence alphabet, either 20 for proteins or four for DNA/RNA. This means that the maximal sequence information content per position is $\log_2 4 = 2\ bits$ for DNA/RNA and $\log_2 20 \approx 4.32\ bits$ for proteins.

The original implementation by Schneider does not handle sequence gaps.

We have slightly modified the algorithm so an estimated logo is presented in areas with sequence gaps.

If amino acid residues or nucleotides of one sequence are found in an area containing gaps, we have chosen to show the particular residue as the fraction of the sequences. Example; if one position in the alignment contain 9 gaps and only one alanine (A) the A represented in the logo has a hight of 0.1.

Other useful resources
The website of Tom Schneider
http://www-lmmb.ncifcrf.gov/~toms/

WebLogo

http://weblogo.berkeley.edu/

[Crooks et al., 2004]

22.3 Edit alignments

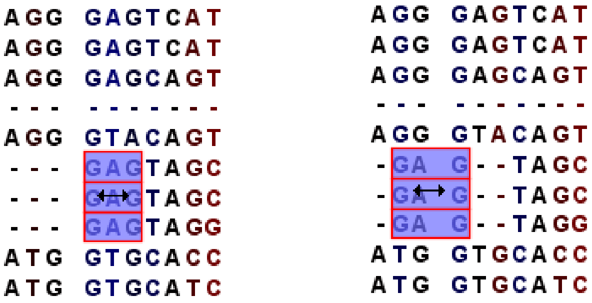
22.3.1 Move residues and gaps

The placement of gaps in the alignment can be changed by modifying the parameters when creating the alignment (see section 22.1). However, gaps and residues can also be moved after the alignment is created:

select one or more gaps or residues in the alignment | drag the selection to move

This can be done both for single sequences, but also for multiple sequences by making a selection covering more than one sequence. When you have made the selection, the mouse pointer turns into a horizontal arrow indicating that the selection can be moved (see figure 22.9).

Note! Residues can only be moved when they are next to a gap.



```
AGG GAGTCAT        AGG GAGTCAT
AGG GAGTCAT        AGG GAGTCAT
AGG GAGCAGT        AGG GAGCAGT
--- -------        --- -------
AGG GTACAGT        AGG GTACAGT
--- GAGTAGC        -GA G--TAGC
--- GAGTAGC        -GA G--TAGC
--- GAGTAGG        -GA G--TAGG
ATG GTGCACC        ATG GTGCACC
ATG GTGCATC        ATG GTGCATC
```

Figure 22.9: *Moving a part of an alignment. Notice the change of mouse pointer to a horizontal arrow.*

22.3.2 Insert gaps

The placement of gaps in the alignment can be changed by modifying the parameters when creating the alignment. However, gaps can also be added manually after the alignment is created.

To insert extra gaps:

select a part of the alignment | right-click the selection | Add gaps before/after

If you have made a selection covering e.g. five residues, a gap of five will be inserted. In this way you can easily control the number of gaps to insert. Gaps will be inserted in the sequences that you selected. If you make a selection in two sequences in an alignment, gaps will be inserted into these two sequences. This means that these two sequences will be displaced compared to the other sequences in the alignment.

22.3.3 Delete residues and gaps

Residues or gaps can be deleted for individual sequences or for the whole alignment. For individual sequences:

select the part of the sequence you want to delete | right-click the selection | Edit Selection ( ) | Delete the text in the dialog | Replace

The selection shown in the dialog will be replaced by the text you enter. If you delete the text, the selection will be replaced by an empty text, i.e. deleted.

To delete entire columns:

select the part of the alignment you want to delete | right-click the selection | Delete columns

The selection may cover one or more sequences, but the **Delete columns** function will always apply to the entire alignment.

22.3.4 Copy annotations to other sequences

Annotations on one sequence can be transferred to other sequences in the alignment:

right-click the annotation | Copy Annotation to other Sequences

This will display a dialog listing all the sequences in the alignment. Next to each sequence is a checkbox which is used for selecting which sequences, the annotation should be copied to. Click **Copy** to copy the annotation.

22.3.5 Move sequences up and down

Sequences can be moved up and down in the alignment:

drag the name of the sequence up or down

When you move the mouse pointer over the label, the pointer will turn into a vertical arrow indicating that the sequence can be moved.

The sequences can also be sorted automatically to let you save time moving the sequences around. To sort the sequences alphabetically:

Right-click the name of a sequence | Sort Sequences Alphabetically

If you change the Sequence name (in the **Sequence Layout** view preferences), you will have to ask the program to sort the sequences again.

The sequences can also be sorted by similarity, grouping similar sequences together:

Right-click the name of a sequence | Sort Sequences by Similarity

22.3.6 Delete, rename and add sequences

Sequences can be removed from the alignment by right-clicking the label of a sequence:

right-click label | Delete Sequence

This can be undone by clicking **Undo** ( ) in the Toolbar.

If you wish to delete several sequences, you can check all the sequences, right-click and choose **Delete Marked Sequences**. To show the checkboxes, you first have to click the **Show Selection Boxes** in the **Side Panel**.

A sequence can also be renamed:

> **right-click label | Rename Sequence**

This will show a dialog, letting you rename the sequence. This will not affect the sequence that the alignment is based on.

Extra sequences can be added to the alignment by creating a new alignment where you select the current alignment and the extra sequences (see section 22.1).

The same procedure can be used for joining two alignments.

22.3.7 Realign selection

If you have created an alignment, it is possible to realign a part of it, leaving the rest of the alignment unchanged:

> **select a part of the alignment to realign | right-click the selection | Realign selection**

This will open **Step 2** in the "Create alignment" dialog, allowing you to set the parameters for the realignment (see section 22.1).

It is possible for an alignment to become shorter or longer as a result of the realignment of a region. This is because gaps may have to be inserted in, or deleted from, the sequences not selected for realignment. This will only occur for entire columns of gaps in these sequences, ensuring that their relative alignment is unchanged.

Realigning a selection is a very powerful tool for editing alignments in several situations:

- **Removing changes.** If you change the alignment in a specific region by hand, you may end up being unhappy with the result. In this case you may of course undo your edits, but another option is to select the region and realign it.
- **Adjusting the number of gaps.** If you have a region in an alignment which has too many gaps in your opinion, you can select the region and realign it. By choosing a relatively high gap cost you will be able to reduce the number of gaps.
- **Combine with fixpoints.** If you have an alignment where two residues are not aligned, but you know that they should have been. You can now set an alignment fixpoint on each of the two residues, select the region and realign it using the fixpoints. Now, the two residues are aligned with each other and everything in the selected region around them is adjusted to accommodate this change.

22.4 Join alignments

CLC Genomics Workbench can join several alignments into one. This feature can for example be used to construct "supergenes" for phylogenetic inference by joining alignments of several disjoint genes into one spliced alignment. Note, that when alignments are joined, all their annotations are carried over to the new spliced alignment.

Alignments can be joined by:

> **select alignments to join | Toolbox in the Menu Bar | Alignments and Trees ( ) | Join Alignments ( )**

or **select alignments to join | right-click either selected alignment | Toolbox | Alignments and Trees (▤) | Join Alignments (▤)**

This opens the dialog shown in figure 22.10.

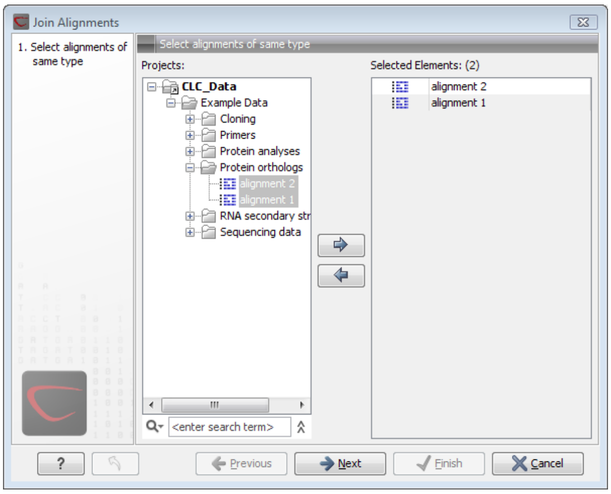


Figure 22.10: *Selecting two alignments to be joined.*

If you have selected some alignments before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove alignments from the selected elements. Click **Next** opens the dialog shown in figure 22.11.

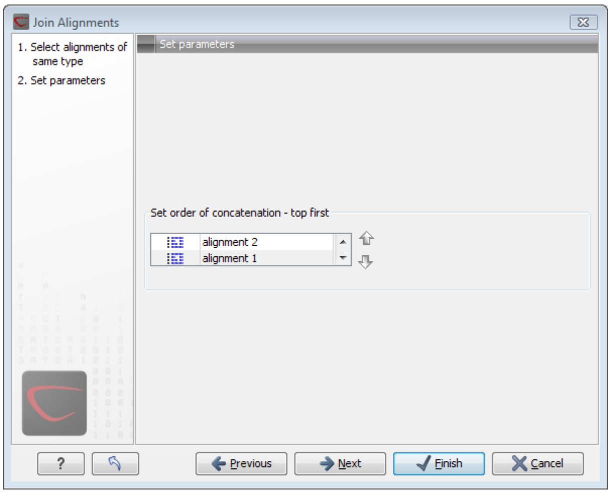


Figure 22.11: *Selecting order of concatenation.*

To adjust the order of concatenation, click the name of one of the alignments, and move it up or down using the arrow buttons.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**. The result is seen in figure 22.12.

22.4.1 How alignments are joined

Alignments are joined by considering the sequence names in the individual alignments. If two sequences from different alignments have identical names, they are considered to have the

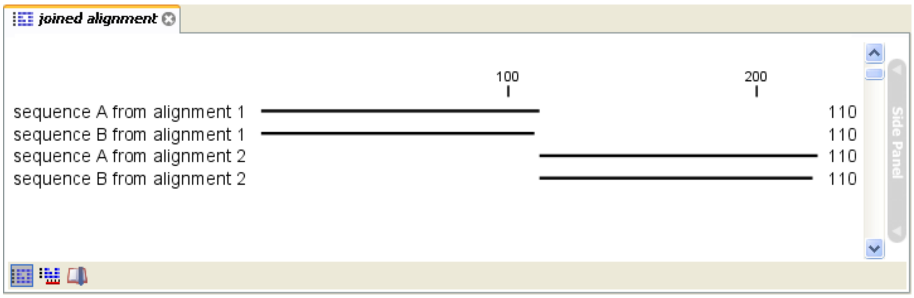

joined alignment
100
200
sequence A from alignment 1
sequence B from alignment 1
sequence A from alignment 2
sequence B from alignment 2
110
110
110
110
Side Panel

Figure 22.12: *The joining of the alignments result in one alignment containing rows of sequences corresponding to the number of uniquely named sequences in the joined alignments.*

same origin and are thus joined. Consider the joining of alignments A and B. If a sequence named "in-A-and-B" is found in both A and B, the spliced alignment will contain a sequence named "in-A-and-B" which represents the characters from A and B joined in direct extension of each other. If a sequence with the name "in-A-not-B" is found in A but not in B, the spliced alignment will contain a sequence named "in-A-not-B". The first part of this sequence will contain the characters from A, but since no sequence information is available from B, a number of gap characters will be added to the end of the sequence corresponding to the number of residues in B. Note, that the function does not require that the individual alignments contain an equal number of sequences.

22.5 Pairwise comparison

For a given set of aligned sequences (see chapter 22) it is possible make a pairwise comparison in which each pair of sequences are compared to each other. This provides an overview of the diversity among the sequences in the alignment.

In *CLC Genomics Workbench* this is done by creating a comparison table:

Toolbox in the Menu Bar | **Alignments and Trees** (▤) | **Pairwise Comparison** (▦)

or **right-click alignment in Navigation Area** | **Toolbox** | **Alignments and Trees** (▤) | **Pairwise Comparison** (▦)

This opens the dialog displayed in figure 22.13:

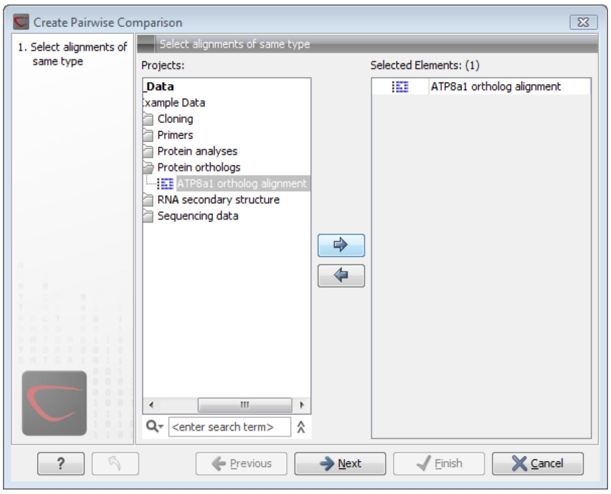


Figure 22.13: *Creating a pairwise comparison table.*

If an alignment was selected before choosing the Toolbox action, this alignment is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove elements from the **Navigation Area**. Click **Next** to adjust parameters.

22.5.1 Pairwise comparison on alignment selection

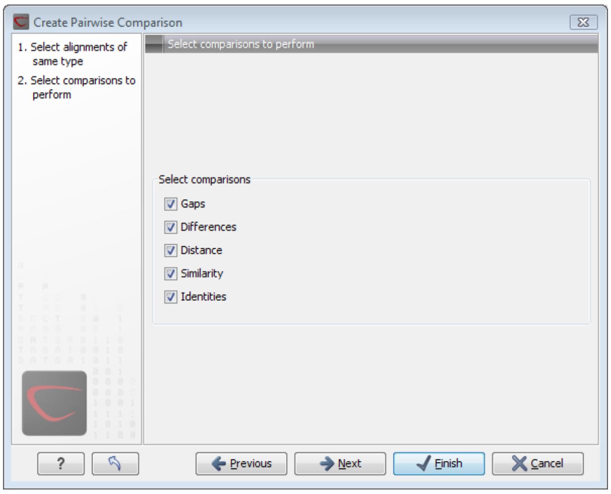
A pairwise comparison can also be performed for a selected part of an alignment:

right-click on an alignment selection | Pairwise Comparison (▦)

This leads directly to the dialog described in the next section.

22.5.2 Pairwise comparison parameters

There are four kinds of comparison that can be made between the sequences in the alignment, as shown in figure 22.14.


Create Pairwise Comparison
1. Select alignments of same type
2. Select comparisons to perform
Select comparisons to perform
?
Previous
Next
Finish
Cancel

Select comparisons

☑ Gaps
☑ Differences
☑ Distance
☑ Similarity
☑ Identities

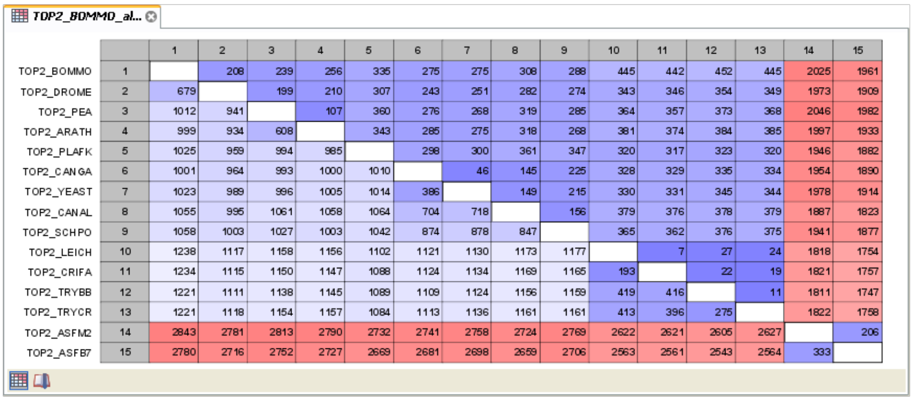
Figure 22.14: *Adjusting parameters for pairwise comparison.*

- **Gaps** Calculates the number of alignment positions where one sequence has a gap and the other does not.
- **Identities** Calculates the number of identical alignment positions to overlapping alignment positions between the two sequences.
- **Differences** Calculates the number of alignment positions where one sequence is different from the other. This includes gap differences as in the Gaps comparison.
- **Distance** Calculates the Jukes-Cantor distance between the two sequences. This number is given as the Jukes-Cantor correction of the proportion between identical and overlapping alignment positions between the two sequences.
- **Percent identity** Calculates the percentage of identical residues in alignment positions to overlapping alignment positions between the two sequences.

Click **Next** if you wish to adjust how to handle the results (see section 9.2). If not, click **Finish**.

22.5.3 The pairwise comparison table

The table shows the results of selected comparisons (see an example in figure 22.15). Since comparisons are often symmetric, the table can show the results of two comparisons at the same time, one in the upper-right and one in the lower-left triangle.



TOP2_BOMMO_al...

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
TOP2_BOMMO	1		208	239	256	335	275	275	308	288	445	442	452	445	2025	1961
TOP2_DROME	2	679		199	210	307	243	251	282	274	343	346	354	349	1973	1909
TOP2_PEA	3	1012	941		107	360	276	268	319	285	364	357	373	368	2046	1982
TOP2_ARATH	4	999	934	608		343	285	275	318	268	381	374	384	385	1997	1933
TOP2_PLAFK	5	1025	959	994	985		298	300	361	347	320	317	323	320	1946	1882
TOP2_CANGA	6	1001	964	993	1000	1010		46	145	225	328	329	335	334	1954	1890
TOP2_YEAST	7	1023	989	996	1005	1014	386		149	215	330	331	345	344	1978	1914
TOP2_CANAL	8	1055	995	1061	1058	1064	704	718		156	379	376	378	379	1887	1823
TOP2_SCHPO	9	1058	1003	1027	1003	1042	874	878	847		365	362	376	375	1941	1877
TOP2_LEICH	10	1238	1117	1158	1156	1102	1121	1130	1173	1177		7	27	24	1818	1754
TOP2_CRIFA	11	1234	1115	1150	1147	1088	1124	1134	1169	1165	193		22	19	1821	1757
TOP2_TRYBB	12	1221	1111	1138	1145	1089	1109	1124	1156	1159	419	416		11	1811	1747
TOP2_TRYCR	13	1221	1118	1154	1157	1084	1113	1136	1161	1161	413	396	275		1822	1758
TOP2_ASFM2	14	2843	2781	2813	2790	2732	2741	2758	2724	2769	2622	2621	2605	2627		206
TOP2_ASFB7	15	2780	2716	2752	2727	2669	2681	2698	2659	2706	2563	2561	2543	2564	333	

Figure 22.15: *A pairwise comparison table.*

The following settings are present in the side panel:

- **Contents**
  - **Upper comparison.** Selects the comparison to show in the upper triangle of the table
  - **Upper comparison gradient.** Selects the color gradient to use for the upper triangle.
  - **Lower comparison** Selects the comparison to show in the lower triangle. Choose the same comparison as in the upper triangle to show all the results of an asymmetric comparison.
  - **Lower comparison gradient.** Selects the color gradient to use for the lower triangle.
  - **Diagonal from upper.** Use this setting to show the diagonal results from the upper comparison.
  - **Diagonal from lower.** Use this setting to show the diagonal results from the lower comparison.
  - **No Diagonal.** Leaves the diagonal table entries blank.
- **Layout**
  - **Lock headers.** Locks the sequence labels and table headers when scrolling the table.
  - **Sequence label.** Changes the sequence labels.
- **Text format**
  - **Text size.** Changes the size of the table and the text within it.

- **Font.** Changes the font in the table.
- **Bold.** Toggles the use of boldface in the table.

22.6 Bioinformatics explained: Multiple alignments

Multiple alignments are at the core of bioinformatical analysis. Often the first step in a chain of bioinformatical analyses is to construct a multiple alignment of a number of homologs DNA or protein sequences. However, despite their frequent use, the development of multiple alignment algorithms remains one of the algorithmically most challenging areas in bioinformatical research.

Constructing a multiple alignment corresponds to developing a hypothesis of how a number of sequences have evolved through the processes of character substitution, insertion and deletion. The input to multiple alignment algorithms is a number of homologous sequences i.e. sequences that share a common ancestor and most often also share molecular function. The generated alignment is a table (see figure 22.16) where each row corresponds to an input sequence and each column corresponds to a position in the alignment. An individual column in this table represents residues that have all diverged from a common ancestral residue. Gaps in the table (commonly represented by a '-') represent positions where residues have been inserted or deleted and thus do not have ancestral counterparts in all sequences.

22.6.1 Use of multiple alignments

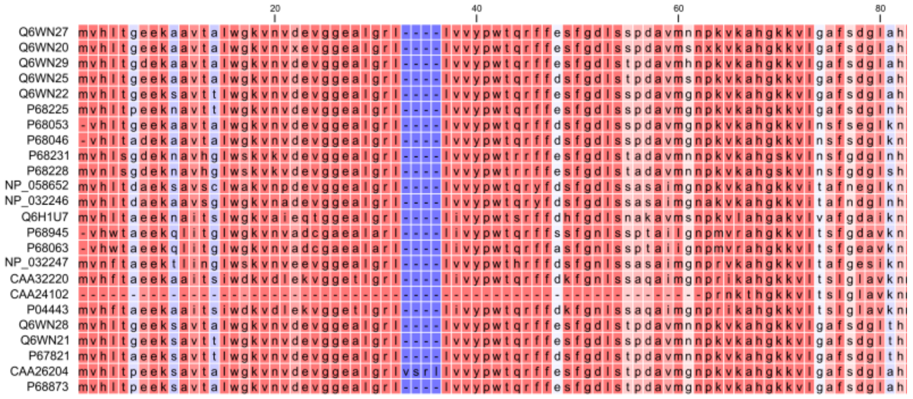
Once a multiple alignment is constructed it can form the basis for a number of analyses:

- The phylogenetic relationship of the sequences can be investigated by tree-building methods based on the alignment.
- Annotation of functional domains, which may only be known for a subset of the sequences, can be transferred to aligned positions in other un-annotated sequences.
- Conserved regions in the alignment can be found which are prime candidates for holding functionally important sites.
- Comparative bioinformatical analysis can be performed to identify functionally important regions.

22.6.2 Constructing multiple alignments

Whereas the optimal solution to the pairwise alignment problem can be found in reasonable time, the problem of constructing a multiple alignment is much harder.

The first major challenge in the multiple alignment procedure is how to rank different alignments i.e. which *scoring function* to use. Since the sequences have a shared history they are correlated through their *phylogeny* and the scoring function should ideally take this into account. Doing so is, however, not straightforward as it increases the number of model parameters considerably. It is therefore commonplace to either ignore this complication and assume sequences to be unrelated, or to use heuristic corrections for shared ancestry.



```
                            20                  40                  60                  80
Q6WN27    mvhltgeekaavtalwgkvnvdevggealgrl----lvvypwtqrffesfgdlsspdavmnnpkvkahgkkvlgafsdglah
Q6WN20    mvhltgeekaavtalwgkvnvxevggealgrl----lvvypwtqrffesfgdlsspdavmsnxkvkahgkkvlgafsdglah
Q6WN29    mvhltgdekaavtalwgkvnvdevggealgrl----lvvypwtqrffesfgdlstpdavmhnpkvkahgkkvlgafsdglah
Q6WN25    mvhltgeekaavtalwgkvnvdevggealgrl----lvvypwtqrffdsfgdlstpdavmsnpkvkahgkkvlgafsdglah
Q6WN22    mvhltgeeksavttlwgkvnvdevggealgrl----lvvypwtqrffesfgdlsspdavmgnpkvkahgkkvlgafsdglah
P68225    mvhltpeeknavttlwgkvnvdevggealgrl----lvvypwtqrffesfgdlsspdavmgnpkvkahgkkvlgafsdglnh
P68053    -vhltgeekaavtalwgkvnvdevggealgrl----lvvypwtqrffdsfgdlsspdavmgnpkvkahgkkvlnsfseglkn
P68046    -vhltadekaavtalwgkvnvdevggealgrl----lvvypwtqrffdsfgdlsspdavmgnpkvkahgkkvlnsfsdglkn
P68231    mvhlsgdeknavhglwskvkvdevggealgrl----lvvypwtrrffesfgdlstadavmnnpkvkahgskvlnsfgdglnh
P68228    mvnlsgdeknavhglwskvkvdevggealgrl----lvvypwtrrffesfgdlstadavmnnpkvkahgskvlnsfgdglsh
NP_058652 mvhltdaeksavsclwakvnpdevggealgrl----lvvypwtqryfdsfgdlssasaimgnpkvkahgkkvitafneglkn
NP_032246 mvhltdaekaavsglwgkvnadevggealgrl----lvvypwtqryfdsfgdlssasaimgnakvkahgkkvitafndglnh
Q6H1U7    mvhltaeeknaitslwgkvaieqtggealgrl----livypwtsrffdhfgdlsnakavmsnpkvlahgakvlvafgdaikn
P68945    -vhwtaeekqlitglwgkvnvadcgaealarl----livypwtqrffssfgnlssptailgnpmvrahgkkvltsfgdavkn
P68063    -vhwtaeekqlitglwgkvnvadcgaealarl----livypwtqrffasfgnlssptaiignpmvrahgkkvltsfgeavkn
NP_032247 mvnftaeektlinglwskvnveevggealgrl----lvvypwthrffdsfgnlssasaimgnprvkahgkkvltafgesikn
CAA32220  mvhftaeekaaitsiwdkvdlekvggetlgrl----livypwtqrffdkfgnlssaqaimgnprikahgkkvltslglavkn
CAA24102  ----------------------------------------------------------------prnkthgkkvltslglavkn
P04443    mvhftaeekaaitsiwdkvdlekvggetlgrl----livypwtqrffdkfgnlssaqaimgnprikahgkkvltslglavkn
Q6WN28    mvhltgeeksavtalwgkvnvdevggealgrl----lvvypwtqrffesfgdlstpdavmnnpkvkahgkkvlgafsdglth
Q6WN21    mvhltgeeksavttlwgkvnvdevggealgrl----lvvypwtqrffesfgdlsspdavmnnpkvkahgkkvlgafsdglth
P67821    mvhltaeeksavttlwgkvnvdevggealgrl----lvvypwtqrffdsfgdlstpdavmnnpkvkahgkkvlgafsdglth
CAA26204  mvhltpeeksavtalwgkvnvdevggealgrlvsrllvvypwtqrffesfgdlstpdavmgnpkvkahgkkvlgafsdglah
P68873    mvhltpeeksavtalwgkvnvdevggealgrl----lvvypwtqrffesfgdlstpdavmgnpkvkahgkkvlgafsdglah
```

Figure 22.16: *The tabular format of a multiple alignment of 24 Hemoglobin protein sequences. Sequence names appear at the beginning of each row and the residue position is indicated by the numbers at the top of the alignment columns. The level of sequence conservation is shown on a color scale with blue residues being the least conserved and red residues being the most conserved.*

The second challenge is to find the optimal alignment given a scoring function. For pairs of sequences this can be done by *dynamic programming* algorithms, but for more than three sequences this approach demands too much computer time and memory to be feasible.

A commonly used approach is therefore to do *progressive alignment* [Feng and Doolittle, 1987] where multiple alignments are built through the successive construction of pairwise alignments. These algorithms provide a good compromise between time spent and the quality of the resulting alignment

Presently, the most exciting development in multiple alignment methodology is the construction of *statistical alignment* algorithms [Hein, 2001], [Hein et al., 2000]. These algorithms employ a scoring function which incorporates the underlying phylogeny and use an explicit stochastic model of molecular evolution which makes it possible to compare different solutions in a statistically rigorous way. The optimization step, however, still relies on dynamic programming and practical use of these algorithms thus awaits further developments.

Creative Commons License

Chapter 23

Phylogenetic trees

Contents

CLC Genomics Workbench offers different ways of inferring phylogenetic trees. The first part of this chapter will briefly explain the different ways of inferring trees in *CLC Genomics Workbench*. The second part, "Bioinformatics explained", will give a more general introduction to the concept of phylogeny and the associated bioinformatics methods.

23.1 Inferring phylogenetic trees

For a given set of aligned sequences (see chapter 22) it is possible to infer their evolutionary relationships. In *CLC Genomics Workbench* this may be done either by using a distance based method (see "Bioinformatics explained" in section 23.2.) or by using the statistically founded maximum likelihood (ML) approach [Felsenstein, 1981]. Both approaches generate a phylogenetic tree. The tools are found in:

Toolbox | Alignments and trees ( )

To generate a distance-based phylogenetic tree choose:

Create Tree ( )

and to generate a maximum likelihood based phylogenetic tree choose:

Maximum Likelihood Phylogeny ( )

In both cases the dialog displayed in figure 23.1 will be opened:

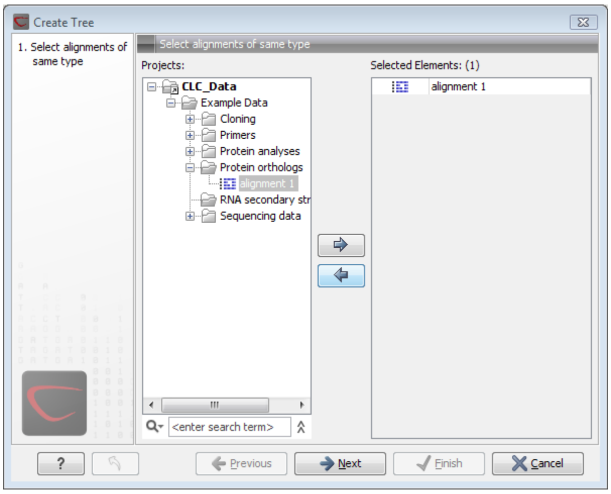
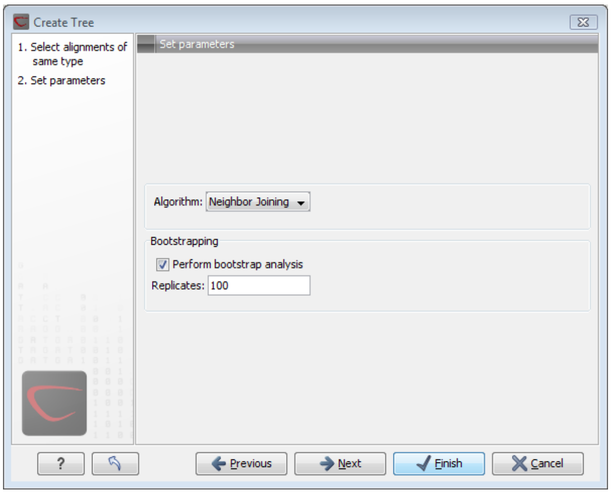


Figure 23.1: *Creating a Tree.*

If an alignment was selected before choosing the Toolbox action, this alignment is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove elements from the **Navigation Area**. Click **Next** to adjust parameters.

23.1.1 Phylogenetic tree parameters

Distance-based methods



Algorithm: Neighbor Joining

Bootstrapping

☑ Perform bootstrap analysis

Replicates: 100

Figure 23.2: *Adjusting parameters for distance-based methods.*

Figure 23.2 shows the parameters that can be set for the distance-based methods:

- Algorithms
  - The **UPGMA** method assumes that evolution has occurred at a constant rate in the different lineages. This means that a root of the tree is also estimated.
  - The **neighbor joining** method builds a tree where the evolutionary rates are free to differ in different lineages. *CLC Genomics Workbench* always draws trees with roots for practical reasons, but with the neighbor joining method, no particular biological hypothesis is postulated by the placement of the root. Figure 23.3 shows the difference between the two methods.

- To evaluate the reliability of the inferred trees, *CLC Genomics Workbench* allows the option of doing a **bootstrap** analysis. A bootstrap value will be attached to each branch, and this value is a measure of the confidence in this branch. The number of replicates in the bootstrap analysis can be adjusted in the wizard. The default value is 100.

For a more detailed explanation, see "Bioinformatics explained" in section 23.2.

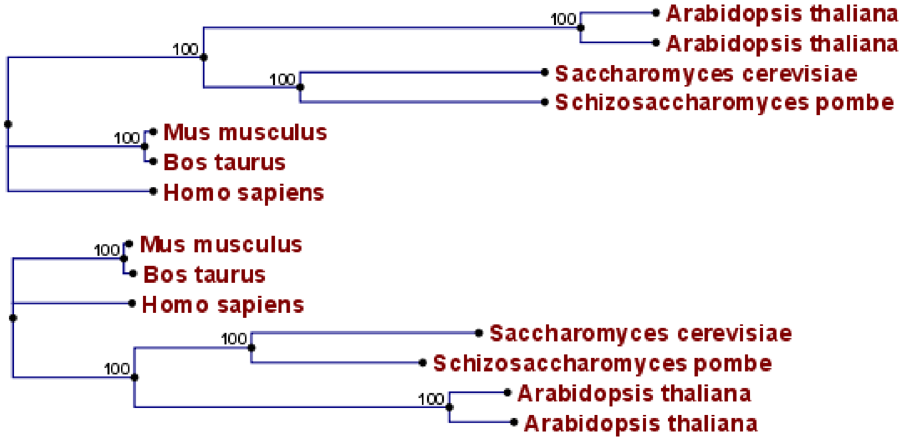
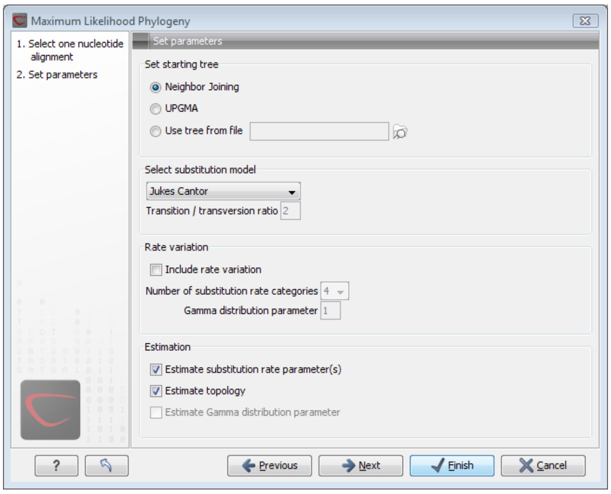


Figure 23.3: *Method choices for phylogenetic inference. The bottom shows a tree found by neighbor joining, while the top shows a tree found by UPGMA. The latter method assumes that the evolution occurs at a constant rate in different lineages.*

Maximum likelihood phylogeny



Maximum Likelihood Phylogeny

1. Select one nucleotide alignment
2. Set parameters

Set parameters

Set starting tree

● Neighbor Joining
○ UPGMA
○ Use tree from file

Select substitution model

Jukes Cantor

Transition / transversion ratio 2

Rate variation

□ Include rate variation

Number of substitution rate categories 4

Gamma distribution parameter 1

Estimation

☑ Estimate substitution rate parameter(s)
☑ Estimate topology
□ Estimate Gamma distribution parameter

Figure 23.4: *Adjusting parameters for ML phylogeny*

Figure 23.4 shows the parameters that can be set for the ML phylogenetic tree reconstruction:

- **Starting tree**: the user is asked to specify a starting tree for the tree reconstruction. There are three possibilities
  - Neighbor joining
  - UPGMA

  - Use tree from file.

- **Select substitution model**: *CLC Genomics Workbench* allows maximum likelihood tree estimation to be performed under the assumption of one of four substitution models: the Jukes Cantor [Jukes and Cantor, 1969], the Kimura 80 [Kimura, 1980], the HKY [Hasegawa et al., 1985] and the GTR (also known as the REV model) [Yang, 1994a] models. All models are time-reversible. The JC and K80 models assume equal base frequencies and the HKY and GTR models allow the frequencies of the four bases to differ (they will be estimated by the observed frequencies of the bases in the alignment). In the JC model all substitutions are assumed to occur at equal rates, in the K80 and HKY models transition and transversion rates are allowed to differ. The GTR model is the general time reversible model and allows all substitutions to occur at different rates. In case of the K80 and HKY models the user may set a transtion/transversion ratio value which will be used as starting value or fixed, depending on the level of estimation chosen by the user (see below). For the substitution rate matrices describing the substitution models we use the parametrization of Yang [Yang, 1994a].

- **Rate variation**: in *CLC Genomics Workbench* substitution rates may be allowed to differ among the individual nucleotide sites in the alignment by selecting the **include rate variation** box. When selected, the discrete gamma model of Yang [Yang, 1994b] is used to model rate variation among sites. The number of categories used in the dicretization of the gamma distribution as well as the gamma distribution parameter may be adjusted by the user (as the gamma distribution is restricted to have mean 1, there is only one parameter in the distribution)

- **Estimation** estimation is done according to the maximum likelihood principle, that is, a search is performed for the values of the free parameters in the model assumed that results in the highest likelihood of the observed alignment [Felsenstein, 1981]. By ticking the **estimate substitution rate parameters** box, maximum likelihood values of the free parameters in the rate matrix describing the assumed substitution model are found. If the **Estimate topology** box is selected, a search in the space of tree topologies for that which best explains the alignment is performed. If left un-ticked, the starting topology is kept fixed at that of the starting tree. The **Estimate Gamma distribution parameter** is active if rate variation has been included in the model and in this case allows estimation of the Gamma distribution parameter to be switched on or off. If the box is left un-ticked, the value is fixed at that given in the **Rate variation** part. In the absence of rate variation estimation of substitution parameters and branch lengths are carried out according to the expectation maximization algorithm [Dempster et al., 1977]. With rate variation the maximization algorithm is performed. The topology space is searched according to the PHYML method [Guindon and Gascuel, 2003], allowing efficient search and estimation of large phylogenies. Branch lengths are given in terms of expected numbers of substitutions per nucleotide site.

23.1.2 Tree View Preferences

The **Tree View** preferences are these:

- **Text format.** Changes the text format for all of the nodes the tree contains.

  - **Text size.** The size of the text representing the nodes can be modified in tiny, small, medium, large or huge.
  - **Font.** Sets the font of the text of all nodes
  - **Bold.** Sets the text bold if enabled.

- **Tree Layout.** Different layouts for the tree.
  - **Node symbol.** Changes the symbol of nodes into box, dot, circle or none if you don‘t want a node symbol.
  - **Layout.** Displays the tree layout as standard or topology.
  - **Show internal node labels**. This allows you to see labels for the internal nodes. Initially, there are no labels, but right-clicking a node allows you to type a label.
  - **Label color.** Changes the color of the labels on the tree nodes.
  - **Branch label color.** Modifies the color of the labels on the branches.
  - **Node color.** Sets the color of all nodes.
  - **Line color.** Alters the color of all lines in the tree.

- **Labels.** Specifies the text to be displayed in the tree.
  - **Nodes.** Sets the annotation of all nodes either to name or to species.
  - **Branches.** Changes the annotation of the branches to bootstrap, length or none if you don‘t want annotation on branches.

Note! Dragging in a tree will change it. You are therefore asked if you want to save this tree when the **Tree View** is closed.

You may select part of a **Tree** by clicking on the nodes that you want to select.

Right-click a selected node opens a menu with the following options:

- Set root above node (defines the root of the tree to be just above the selected node).
- Set root at this node (defines the root of the tree to be at the selected node).
- Toggle collapse (collapses or expands the branches below the node).
- Change label (allows you to label or to change the existing label of a node).
- Change branch label (allows you to change the existing label of a branch).

You can also relocate leaves and branches in a tree or change the length. It is possible to modify the text on the unit measurement at the bottom of the tree view by right-clicking the text. In this way you can specify a unit, e.g. "years".

Branch lengths are given in terms of expected numbers of substitutions per site.

Note! To drag branches of a tree, you must first click the node one time, and then click the node again, and this time hold the mouse button.

In order to change the representation:

- Rearrange leaves and branches by

  Select a leaf or branch | Move it up and down (Hint: The mouse turns into an arrow pointing up and down)

- Change the length of a branch by

  Select a leaf or branch | Press Ctrl | Move left and right (Hint: The mouse turns into an arrow pointing left and right)

Alter the preferences in the **Side Panel** for changing the presentation of the tree.

23.2 Bioinformatics explained: phylogenetics

Phylogenetics describes the taxonomical classification of organisms based on their evolutionary history i.e. their *phylogeny*. Phylogenetics is therefore an integral part of the science of *systematics* that aims to establish the phylogeny of organisms based on their characteristics. Furthermore, phylogenetics is central to evolutionary biology as a whole as it is the condensation of the overall paradigm of how life arose and developed on earth.

23.2.1 The phylogenetic tree

The evolutionary hypothesis of a phylogeny can be graphically represented by a phylogenetic tree.

Figure 23.5 shows a proposed phylogeny for the great apes, *Hominidae*, taken in part from Purvis [Purvis, 1995]. The tree consists of a number of nodes (also termed vertices) and branches (also termed edges). These nodes can represent either an individual, a species, or a higher grouping and are thus broadly termed taxonomical units. In this case, the terminal nodes (also called leaves or tips of the tree) represent extant species of *Hominidae* and are the *operational taxonomical units* (OTUs). The internal nodes, which here represent extinct common ancestors of the great apes, are termed *hypothetical taxonomical units* since they are not directly observable.

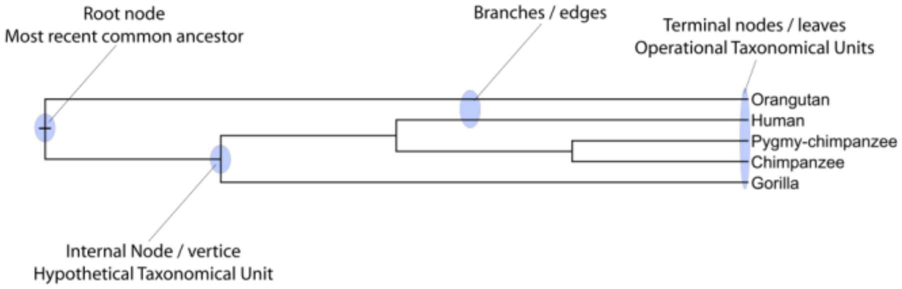


Figure 23.5: *A proposed phylogeny of the great apes (Hominidae). Different components of the tree are marked, see text for description.*

The ordering of the nodes determine the tree *topology* and describes how lineages have diverged over the course of evolution. The branches of the tree represent the amount of evolutionary divergence between two nodes in the tree and can be based on different measurements. A tree is completely specified by its topology and the set of all edge lengths.

The phylogenetic tree in figure 23.5 is rooted at the most recent common ancestor of all *Hominidae* species, and therefore represents a hypothesis of the direction of evolution e.g. that the common ancestor of gorilla, chimpanzee and man existed before the common ancestor of chimpanzee and man. If this information is absent trees can be drawn as unrooted.

23.2.2 Modern usage of phylogenies

Besides evolutionary biology and systematics the inference of phylogenies is central to other areas of research.

As more and more genetic diversity is being revealed through the completion of multiple genomes, an active area of research within bioinformatics is the development of comparative machine learning algorithms that can simultaneously process data from multiple species [Siepel and Haussler, 2004]. Through the comparative approach, valuable evolutionary information can be obtained about which amino acid substitutions are functionally tolerant to the organism and which are not. This information can be used to identify substitutions that affect protein function and stability, and is of major importance to the study of proteins [Knudsen and Miyamoto, 2001]. Knowledge of the underlying phylogeny is, however, paramount to comparative methods of inference as the phylogeny describes the underlying correlation from shared history that exists between data from different species.

In molecular epidemiology of infectious diseases, phylogenetic inference is also an important tool. The very fast substitution rate of microorganisms, especially the RNA viruses, means that these show substantial genetic divergence over the time-scale of months and years. Therefore, the phylogenetic relationship between the pathogens from individuals in an epidemic can be resolved and contribute valuable epidemiological information about transmission chains and epidemiologically significant events [Leitner and Albert, 1999], [Forsberg et al., 2001].

23.2.3 Reconstructing phylogenies from molecular data

Traditionally, phylogenies have been constructed from morphological data, but following the growth of genetic information it has become common practice to construct phylogenies based on molecular data, known as *molecular phylogeny*. The data is most commonly represented in the form of DNA or protein sequences, but can also be in the form of e.g. restriction fragment length polymorphism (RFLP).

Methods for constructing molecular phylogenies can be distance based or character based.

Distance based methods
Two common algorithms, both based on pairwise distances, are the UPGMA and the Neighbor Joining algorithms. Thus, the first step in these analyses is to compute a matrix of pairwise distances between OTUs from their sequence differences. To correct for multiple substitutions it is common to use distances corrected by a model of molecular evolution such as the Jukes-Cantor model [Jukes and Cantor, 1969].

UPGMA. A simple but popular clustering algorithm for distance data is Unweighted Pair Group Method using Arithmetic averages (UPGMA) ([Michener and Sokal, 1957], [Sneath and Sokal, 1973]). This method works by initially having all sequences in separate clusters and continuously joining these. The tree is constructed by considering all initial clusters as leaf nodes in the tree, and each time two clusters are joined, a node is added to the tree as the parent of the two chosen nodes. The clusters to be joined are chosen as those with minimal pairwise distance. The branch lengths are set corresponding to the distance between clusters, which is calculated as the average distance between pairs of sequences in each cluster.

The algorithm assumes that the distance data has the so-called *molecular clock* property i.e. the divergence of sequences occur at the same constant rate at all parts of the tree. This means that the leaves of UPGMA trees all line up at the extant sequences and that a root is estimated as part of the procedure.

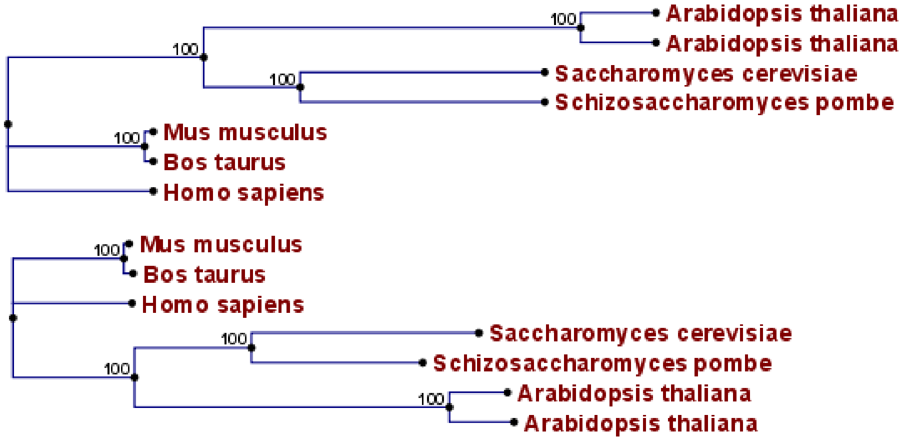


Figure 23.6: *Algorithm choices for phylogenetic inference. The bottom shows a tree found by the neighbor joining algorithm, while the top shows a tree found by the UPGMA algorithm. The latter algorithm assumes that the evolution occurs at a constant rate in different lineages.*

Neighbor Joining. The neighbor joining algorithm, [Saitou and Nei, 1987], on the other hand, builds a tree where the evolutionary rates are free to differ in different lineages, i.e., the tree does not have a particular root. Some programs always draw trees with roots for practical reasons, but for neighbor joining trees, no particular biological hypothesis is postulated by the placement of the root. The method works very much like UPGMA. The main difference is that instead of using pairwise distance, this method subtracts the distance to all other nodes from the pairwise distance. This is done to take care of situations where the two closest nodes are not neighbors in the "real" tree. The neighbor join algorithm is generally considered to be fairly good and is widely used. Algorithms that improves its cubic time performance exist. The improvement is only significant for quite large datasets.

Character based methods. Whereas the distance based methods compress all sequence information into a single number, the character based methods attempt to infer the phylogeny based on all the individual characters (nucleotides or amino acids).

Parsimony. In parsimony based methods a number of sites are defined which are informative about the topology of the tree. Based on these, the best topology is found by minimizing the number of substitutions needed to explain the informative sites. Parsimony methods are not based on explicit evolutionary models.

Maximum Likelihood. Maximum likelihood and Bayesian methods (see below) are probabilistic methods of inference. Both have the pleasing properties of using explicit models of molecular evolution and allowing for rigorous statistical inference. However, both approaches are very computer intensive.

A stochastic model of molecular evolution is used to assign a probability (likelihood) to each phylogeny, given the sequence data of the OTUs. Maximum likelihood inference [Felsenstein, 1981] then consists of finding the tree which assign the highest probability to the data.

Bayesian inference. The objective of Bayesian phylogenetic inference is not to infer a single "correct" phylogeny, but rather to obtain the full posterior probability distribution of all possible phylogenies. This is obtained by combining the likelihood and the prior probability distribution of evolutionary parameters. The vast number of possible trees means that bayesian phylogenetics must be performed by approximative Monte Carlo based methods. [Larget and Simon, 1999], [Yang and Rannala, 1997].

23.2.4 Interpreting phylogenies

Bootstrap values

A popular way of evaluating the reliability of an inferred phylogenetic tree is bootstrap analysis. The first step in a bootstrap analysis is to re-sample the alignment columns with replacement. I.e., in the re-sampled alignment, a given column in the original alignment may occur two or more times, while some columns may not be represented in the new alignment at all. The re-sampled alignment represents an estimate of how a different set of sequences from the same genes and the same species may have evolved on the same tree.

If a new tree reconstruction on the re-sampled alignment results in a tree similar to the original one, this increases the confidence in the original tree. If, on the other hand, the new tree looks very different, it means that the inferred tree is unreliable. By re-sampling a number of times it is possibly to put reliability weights on each internal branch of the inferred tree. If the data was bootstrapped a 100 times, a bootstrap score of 100 means that the corresponding branch occurs in all 100 trees made from re-sampled alignments. Thus, a high bootstrap score is a sign of greater reliability.

Other useful resources

The Tree of Life web-project
http://tolweb.org

Joseph Felsensteins list of phylogeny software
http://evolution.genetics.washington.edu/phylip/software.html

Creative Commons License

Chapter 24

RNA structure

Contents

Ribonucleic acid (RNA) is a nucleic acid polymer that plays several important roles in the cell.

As for proteins, the three dimensional shape of an RNA molecule is important for its molecular function. A number of tertiary RNA structures are know from crystallography but de novo prediction of tertiary structures is not possible with current methods. However, as for proteins RNA tertiary structures can be characterized by secondary structural elements which are hydrogen bonds within the molecule that form several recognizable "domains" of secondary structure like stems, hairpin loops, bulges and internal loops. A large part of the functional information is thus contained in the secondary structure of the RNA molecule, as shown by the high degree of base-pair conservation observed in the evolution of RNA molecules.

Computational prediction of RNA secondary structure is a well defined problem and a large body of work has been done to refine prediction algorithms and to experimentally estimate the relevant biological parameters.

In *CLC Genomics Workbench* we offer the user a number of tools for analyzing and displaying RNA structures. These include:

- Secondary structure prediction using state-of-the-art algorithms and parameters
- Calculation of full partition function to assign probabilities to structural elements and hypotheses
- Scanning of large sequences to find local structure signal
- Inclusion of experimental constraints to the folding process
- Advanced viewing and editing of secondary structures and structure information

24.1 RNA secondary structure prediction

CLC Genomics Workbench uses a minimum free energy (MFE) approach to predict RNA secondary structure. Here, the stability of a given secondary structure is defined by the amount of free energy used (or released) by its formation. The more negative free energy a structure has, the more likely is its formation since more stored energy is released by the event. Free energy contributions are considered additive, so the total free energy of a secondary structure can be calculated by adding the free energies of the individual structural elements. Hence, the task of the prediction algorithm is to find the secondary structure with the minimum free energy. As input to the algorithm empirical energy parameters are used. These parameters summarize the free energy contribution associated with a large number of structural elements. A detailed structure overview can be found in 24.5.

In *CLC Genomics Workbench*, structures are predicted by a modified version of Professor Michael Zukers well known algorithm [Zuker, 1989b] which is the algorithm behind a number of RNA-folding packages including MFOLD. Our algorithm is a dynamic programming algorithm for free energy minimization which includes free energy increments for coaxial stacking of stems when they are either adjacent or separated by a single mismatch. The thermodynamic energy parameters used are from the latest Mfold version 3, see `http://www.bioinfo.rpi.edu/~zukerm/rna/energy/`.

24.1.1 Selecting sequences for prediction

Secondary structure prediction can be accessed in the **Toolbox**:

Toolbox | **RNA Structure** ( ) | **Predict Secondary Structure** ( )

This opens the dialog shown in figure 24.1.

If you have selected sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or

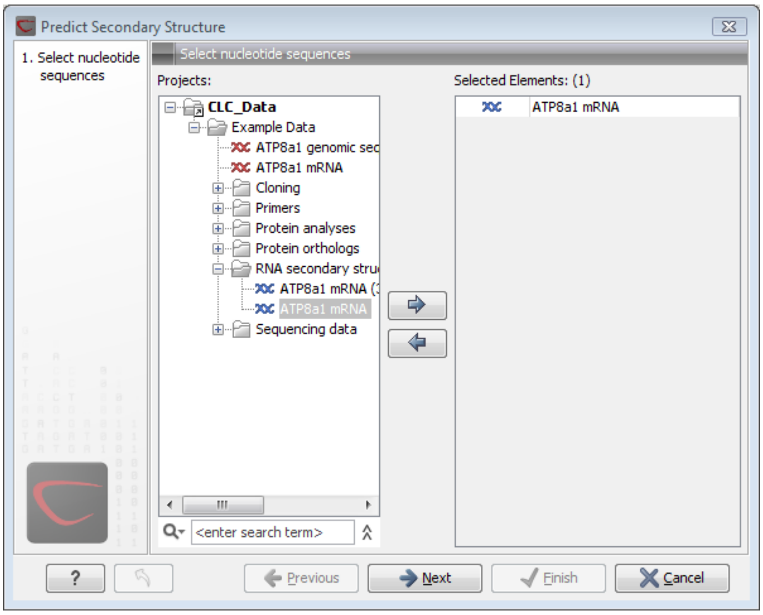
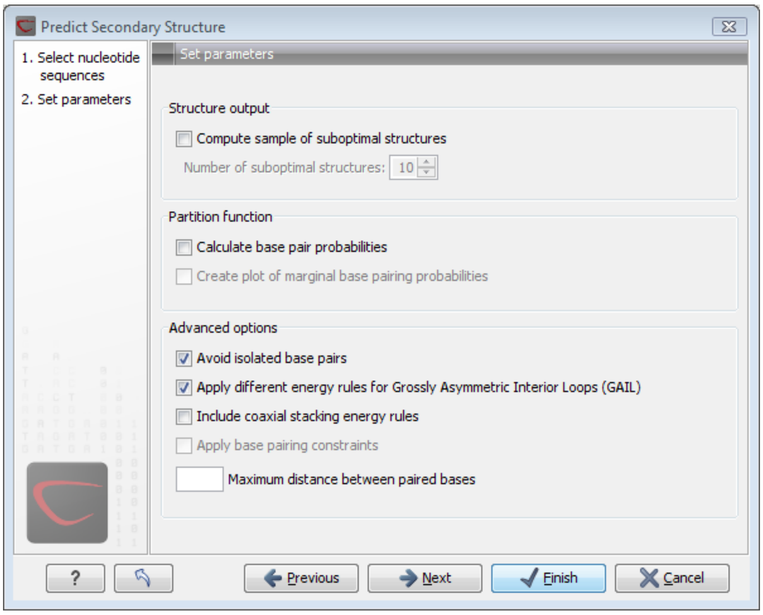


Figure 24.1: *Selecting RNA or DNA sequences for structure prediction (DNA is folded as if it were RNA).*

sequence lists from the selected elements. You can use both DNA and RNA sequences - DNA will be folded as if it were RNA. Click **Next** to adjust secondary structure prediction parameters. Clicking **Next** opens the dialog shown in figure 24.2.



Structure output

□ Compute sample of suboptimal structures

Number of suboptimal structures: 10

Partition function

□ Calculate base pair probabilities

□ Create plot of marginal base pairing probabilities

Advanced options

☑ Avoid isolated base pairs

☑ Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL)

□ Include coaxial stacking energy rules

□ Apply base pairing constraints

Maximum distance between paired bases

Figure 24.2: *Adjusting parameters for secondary structure prediction.*

24.1.2 Structure output

The predict secondary structure algorithm always calculates the minimum free energy structure of the input sequence. In addition to this, it is also possible to compute a sample of suboptimal structures by ticking the checkbox labeled **Compute sample of suboptimal structures**. Subsequently, you can specify how many structures to include in the output. The algorithm then iterates over all permissible canonical base pairs and computes the minimum free energy and associated secondary structure constrained to contain a specified base pair. These structures are then sorted by their minimum free energy and the most optimal are reported given the specified number of structures. Note, that two different sub-optimal structures can have the same minimum free energy. Further information about suboptimal folding can be found in [Zuker, 1989a].

24.1.3 Partition function

The predicted minimum free energy structure gives a point-estimate of the structural conformation of an RNA molecule. However, this procedure implicitly assumes that the secondary structure is at equilibrium, that there is only a single accessible structure conformation, and that the parameters and model of the energy calculation are free of errors.

Obvious deviations from these assumptions make it clear that the predicted MFE structure may deviate somewhat from the actual structure assumed by the molecule. This means that rather than looking at the MFE structure it may be informative to inspect statistical properties of the structural landscape to look for general structural properties which seem to be robust to minor variations in the total free energy of the structure (see [Mathews et al., 2004]).

To this end *CLC Genomics Workbench* allows the user to calculate the complete secondary structure partition function using the algorithm described in [Mathews et al., 2004] which is an extension of the seminal work by [McCaskill, 1990].

There are two options regarding the partition function calculation:

- **Calculate base pair probabilities.** This option invokes the partition function calculation and calculates the marginal probabilities of all possible base pairs and the the marginal probability that any single base is unpaired.
- **Create plot of marginal base pairing probabilities.** This creates a plot of the marginal base pair probability of all possible base pairs as shown in figure 24.3.

The marginal probabilities of base pairs and of bases being unpaired are distinguished by colors which can be displayed in the normal sequence view using the **Side Panel** - see section 24.2.3 and also in the secondary structure view. An example is shown in figure 24.4. Furthermore, the marginal probabilities are accessible from tooltips when hovering over the relevant parts of the structure.

24.1.4 Advanced options

The free energy minimization algorithm includes a number of advanced options:

- **Avoid isolated base pairs**. The algorithm filters out isolated base pairs (i.e. stems of length 1).
- **Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL)**. Compute the minimum free energy applying different rules for Grossly Asymmetry Interior Loops (GAIL). A Grossly Asymmetry Interior Loop (GAIL) is an interior loop that is $1 \times n$ or $n \times 1$ where $n > 2$ (see http://www.bioinfo.rpi.edu/~zukerm/lectures/RNAfold-html/rnafold-print.pdf).

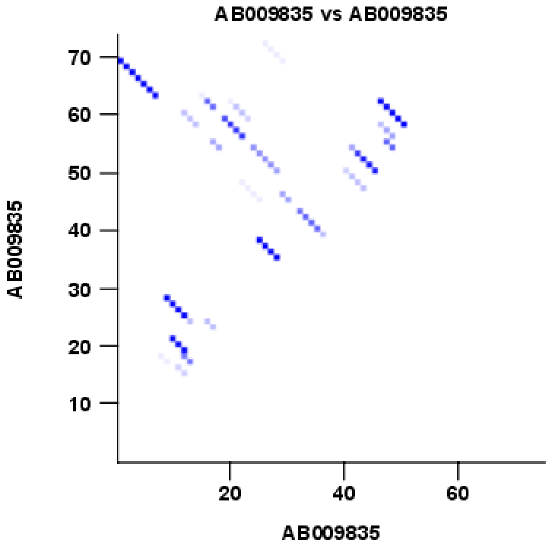


Figure 24.3: *The marginal base pair probability of all possible base pairs.*

- **Include coaxial stacking energy rules**. Include free energy increments of coaxial stacking for adjacent helices [Mathews et al., 2004].
- **Apply base pairing constraints**. With base pairing constraints, you can easily add experimental constraints to your folding algorithm. When you are computing suboptimal structures, it is not possible to apply base pair constraints. The possible base pairing constraints are:
  - Force two equal length intervals to form a stem.
  - Prohibit two equal length intervals to form a stem.
  - Prohibit all nucleotides in a selected region to be a part of a base pair.

  Base pairing constraints have to be added to the sequence before you can use this option - see below.
- **Maximum distance between paired bases**. Forces the algorithms to only consider RNA structures of a given upper length by setting a maximum distance between the base pair that opens a structure.

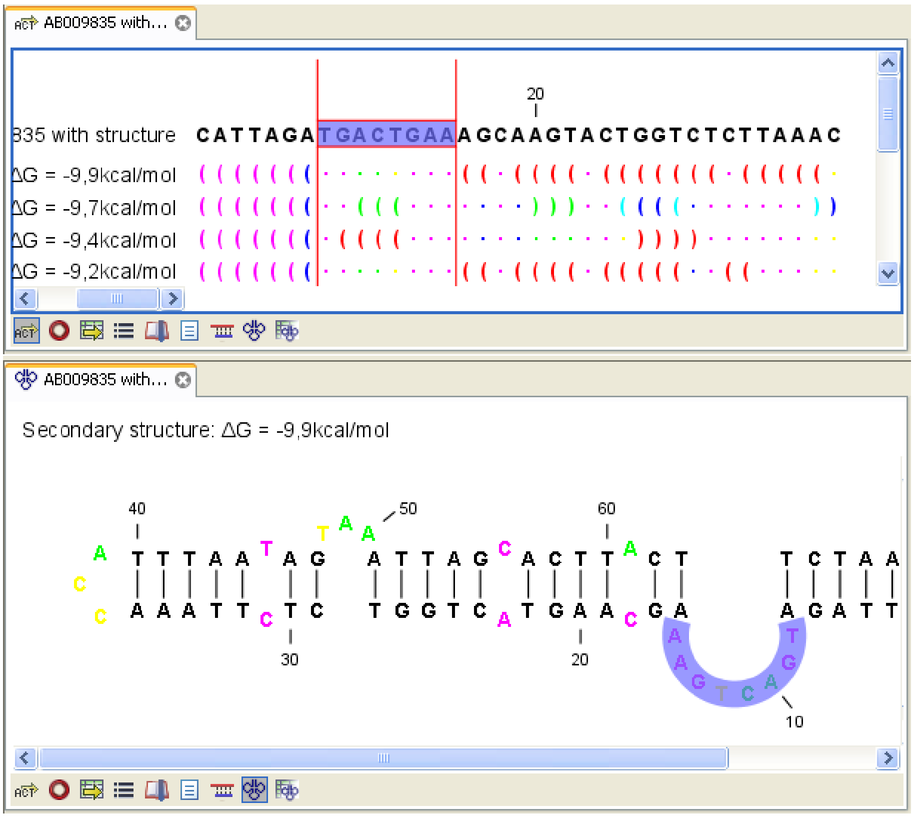
Specifying structure constraints

Structure constraints can serve two purposes in *CLC Genomics Workbench*: they can act as experimental constraints imposed on the MFE structure prediction algorithm or they can form a structure hypothesis to be evaluated using the partition function (see section 24.1.3).

To *force* two regions to form a stem, open a normal sequence view and:

> **Select the two regions you want to force by pressing Ctrl while selecting - (use ⌘ on Mac) | right-click the selection | Add Structure Prediction Constraints| Force Stem Here**

This will add an annotation labeled "Forced Stem" to the sequence (see figure 24.5).



AB009835 with...

835 with structure CATTAGATGACTGAAAGCAAGTACTGGTCTCTTAAAC

ΔG = -9,9kcal/mol

ΔG = -9,7kcal/mol

ΔG = -9,4kcal/mol

ΔG = -9,2kcal/mol

AB009835 with...

Secondary structure: ΔG = -9,9kcal/mol

Figure 24.4: *Marginal probability of base pairs shown in linear view (top) and marginal probability of being unpaired shown in the secondary structure 2D view (bottom).*

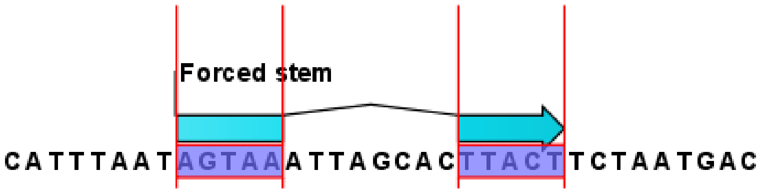


Figure 24.5: *Force a stem of the selected bases.*

Using this procedure to add base pairing constraints will force the algorithm to compute minimum free energy and structure with a stem in the selected region. The two regions must be of equal length.

To *prohibit* two regions to form a stem, open the sequence and:

> **Select the two regions you want to prohibit by pressing Ctrl while selecting - (use ⌘ on Mac) | right-click the selection | Add Structure Prediction Constraints | Prohibit Stem Here**

This will add an annotation labeled "Prohibited Stem" to the sequence (see figure 24.6).

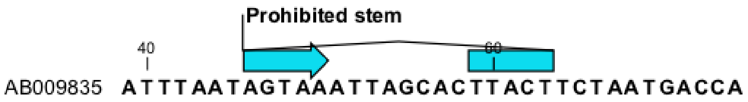


Figure 24.6: *Prohibit the selected bases from forming a stem.*

Using this procedure to add base pairing constraints will force the algorithm to compute minimum free energy and structure without a stem in the selected region. Again, the two selected regions must be of equal length.

To prohibit a region to be part of *any* base pair, open the sequence and:

> **Select the bases you don't want to base pair | right-click the selection | Add Structure Prediction Constraints | Prohibit From Forming Base Pairs**

This will add an annotation labeled "No base pairs" to the sequence, see 24.7.

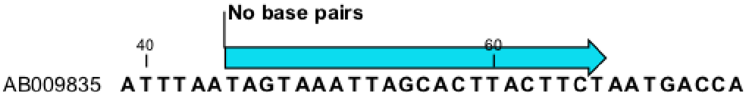


Figure 24.7: *Prohibiting any of the selected base from pairing with other bases.*

Using this procedure to add base pairing constraints will force the algorithm to compute minimum free energy and structure without a base pair containing any residues in the selected region.

When you click **Predict secondary structure** (⚘) and click **Next**, check **Apply base pairing constraints** in order to force or prohibit stem regions or prohibit regions from forming base pairs.

You can add multiple base pairing constraints, e.g. simultaneously adding forced stem regions and prohibited stem regions and prohibit regions from forming base pairs.

24.1.5 Structure as annotation

You can choose to add the elements of the best structure as annotations (see figure 24.8).

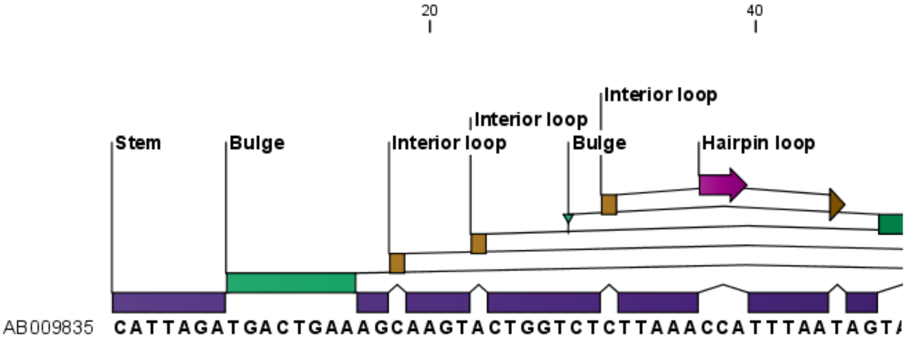


Figure 24.8: *Annotations added for each structure element.*

This makes it possible to use the structure information in other analysis in the *CLC Genomics Workbench*. You can e.g. align different sequences and compare their structure predictions.

Note that possibly existing structure annotation will be removed when a new structure is calculated and added as annotations.

If you generate multiple structures, only the best structure will be added as annotations. If you wish to add one of the sub-optimal structures as annotations, this can be done from the **Show Secondary Structure Table** (▦) described in section 24.2.2.

24.2 View and edit secondary structures

When you predict RNA secondary structure (see section 24.1), the resulting predictions are attached to the sequence and can be shown as:

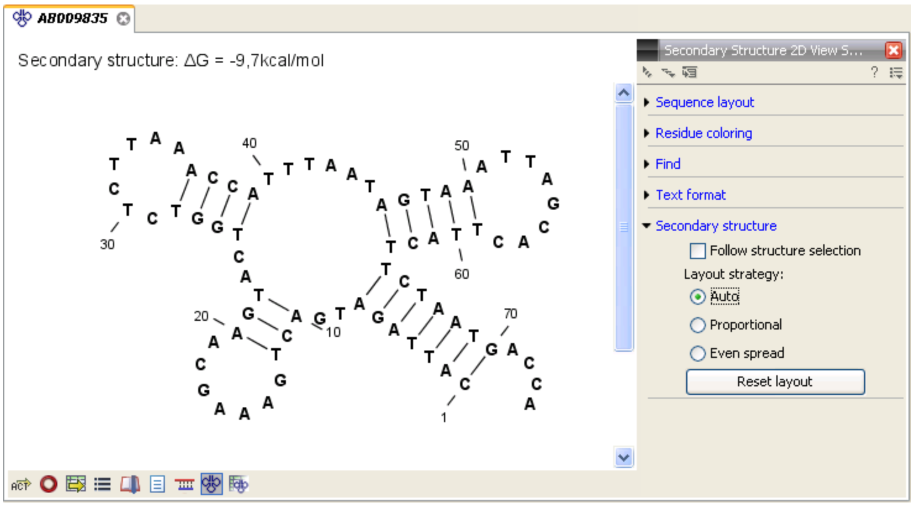
- Annotations in the ordinary sequence views (Linear sequence view (), Annotation table () etc. This is only possible if this has been chosen in the dialog in figure 24.2. See an example in figure 24.8.
- Symbolic representation below the sequence (see section 24.2.3).
- A graphical view of the secondary structure (see section 24.2.1).
- A tabular view of the energy contributions of the elements in the structure. If more than one structure have been predicted, the table is also used to switch between the structures shown in the graphical view. The table is described in section 24.2.2.

24.2.1 Graphical view and editing of secondary structure

To show the secondary view of an already open sequence, click the **Show Secondary Structure 2D View** () button at the bottom of the sequence view.

If the sequence is not open, click **Show** () and select **Secondary Structure 2D View** ().

This will open a view similar to the one shown in figure 24.9.



Secondary structure: ΔG = -9,7kcal/mol

Layout strategy: ● Auto

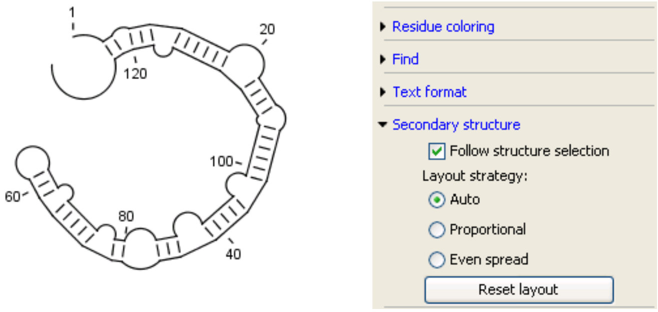
Figure 24.9: *The secondary structure view of an RNA sequence zoomed in.*

Like the normal sequence view, you can use **Zoom in** () and **Zoom out** (). Zooming in will reveal the residues of the structure as shown in figure 24.9. For large structures, zooming out will give you an overview of the whole structure.

Side Panel settings

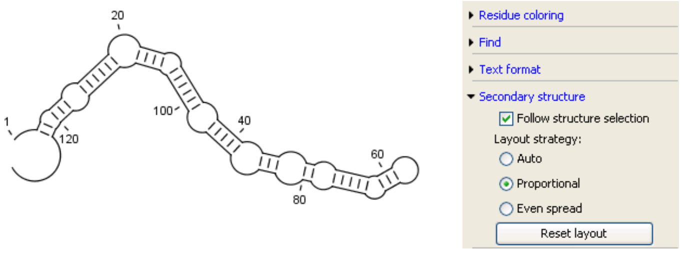
The settings in the **Side Panel** are a subset of the settings in the normal sequence view described in section 10.1.1. However, there are two additional groups of settings unique to the secondary structure 2D view: **Secondary structure**.

- **Follow structure selection.** This setting pertains to the connection between the structures in the secondary structure table ( ) . If this option is checked, the structure displayed in the secondary structure 2D view will follow the structure selections made in this table. See section 24.2.2 for more information.
- **Layout strategy**. Specify the strategy used for the layout of the structure. In addition to these strategies, you can also modify the layout manually as explained in the next section.
  - **Auto.** The layout is adjusted to minimize overlapping structure elements [Han et al., 1999]. This is the default setting (see figure 24.10).
  - **Proportional.** Arc lengths are proportional to the number of residues (see figure 24.11). Nothing is done to prevent overlap.
  - **Even spread.** Stems are spread evenly around loops as shown in figure 24.12.
- **Reset layout.** If you have manually modified the layout of the structure, clicking this button will reset the structure to the way it was laid out when it was created.



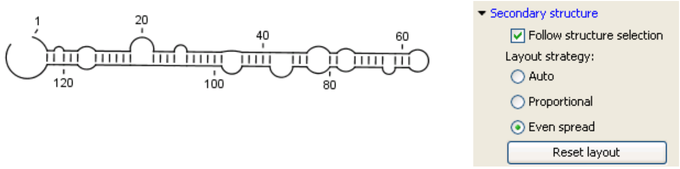
☑ Follow structure selection
Layout strategy:
● Auto
○ Proportional
○ Even spread

Figure 24.10: *Auto layout. Overlaps are minimized.*



☑ Follow structure selection
Layout strategy:
○ Auto
● Proportional
○ Even spread

Figure 24.11: *Proportional layout. Length of the arc is proportional to the number of residues in the arc.*



☑ Follow structure selection
Layout strategy:
○ Auto
○ Proportional
● Even spread

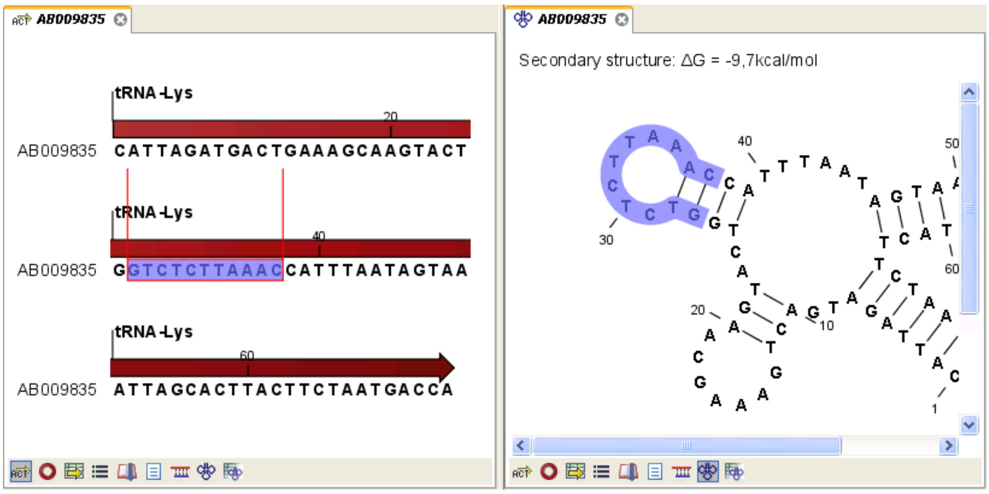
Figure 24.12: *Even spread. Stems are spread evenly around loops.*

Selecting and editing

When you are in **Selection mode** ( ), you can select parts of the structure like in a normal sequence view:

Press down the mouse button where the selection should start | move the mouse cursor to where the selection should end | release the mouse button

One of the advantages of the secondary structure 2D view is that it is integrated with other views of the same sequence. This means that any selection made in this view will be reflected in other views (see figure 24.13).



AB009835 CATTAGATGACTGAAAGCAAGTACT

AB009835 GGTCTCTTAAACCATTTAATAGTAA

AB009835 ATTAGCACTTACTTCTAATGACCA

Secondary structure: ΔG = -9,7kcal/mol

Figure 24.13: *A split view of the secondary structure view and a linear sequence view.*

If you make a selection in another sequence view, this will will also be reflected in the secondary structure view.

The *CLC Genomics Workbench* seeks to produce a layout of the structure where none of the elements overlap. However, it may be desirable to manually edit the layout of a structure for ease of understanding or for the purpose of publication.

To edit a structure, first select the **Pan** (✋) mode in the Tool bar. Now place the mouse cursor on the opening of a stem, and a visual indication of the anchor point for turning the substructure will be shown (see figure 24.14).

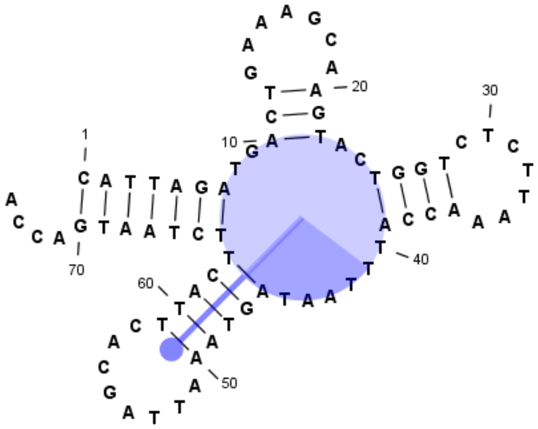


Figure 24.14: *The blue circle represents the anchor point for rotating the substructure.*

Click and drag to rotate the part of the structure represented by the line going from the anchor point. In order to keep the bases in a relatively sequential arrangement, there is a restriction on how much the substructure can be rotated. The highlighted part of the circle represents the angle where rotating is allowed.

In figure 24.15, the structure shown in figure 24.14 has been modified by dragging with the mouse.

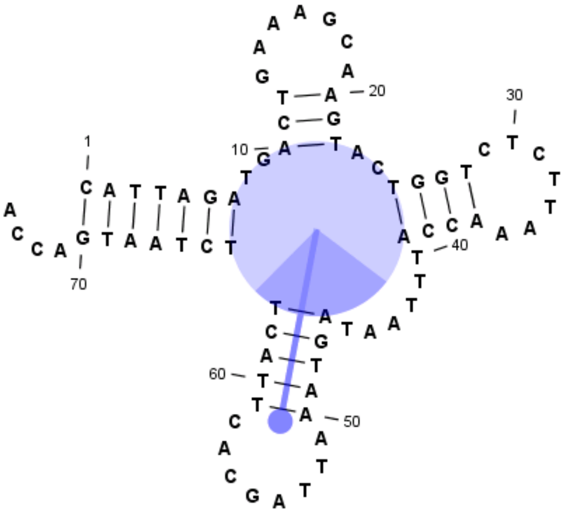


Figure 24.15: *The structure has now been rotated.*

Press **Reset layout** in the **Side Panel** to reset the layout to the way it looked when the structure was predicted.

24.2.2 Tabular view of structures and energy contributions

There are three main reasons to use the **Secondary structure table**:

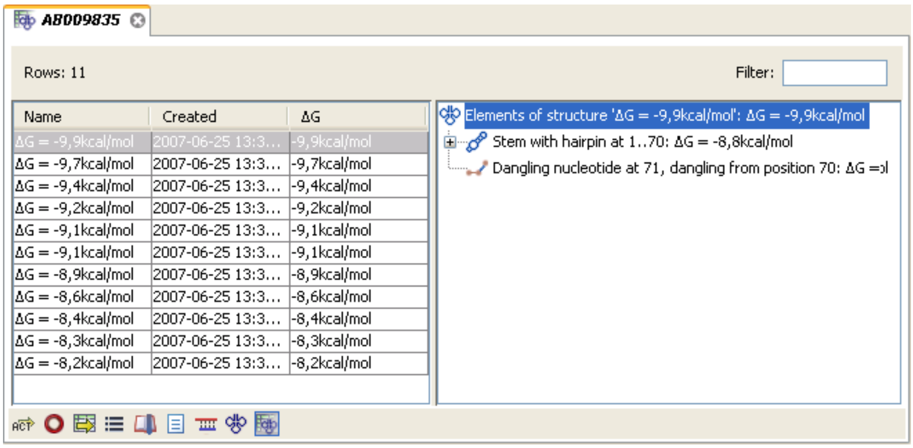
- If more than one structure is predicted (see section 24.1), the table provides an overview of all the structures which have been predicted.
- With multiple structures you can use the table to determine which structure should be displayed in the Secondary structure 2D view (see section 24.2.1).
- The table contains a hierarchical display of the elements in the structure with detailed information about each element's energy contribution.

To show the secondary structure table of an already open sequence, click the **Show Secondary Structure Table** ( ) button at the bottom of the sequence view.

If the sequence is not open, click **Show** ( ) and select **Secondary Structure Table** ( ).

This will open a view similar to the one shown in figure 24.16.

On the left side, all computed structures are listed with the information about structure name, when the structure was created, the free energy of the structure and the probability of the structure if the partition function was calculated. Selecting a row (equivalent: a structure) will display a tree of the contained substructures with their contributions to the total structure free energy. Each substructure contains a union of nested structure elements and other substructures (see a detailed description of the different structure elements in section 24.5.2). Each substructure



Rows: 11

Filter:

Name	Created	ΔG
ΔG = -9,9kcal/mol	2007-06-25 13:3...	-9,9kcal/mol
ΔG = -9,7kcal/mol	2007-06-25 13:3...	-9,7kcal/mol
ΔG = -9,4kcal/mol	2007-06-25 13:3...	-9,4kcal/mol
ΔG = -9,2kcal/mol	2007-06-25 13:3...	-9,2kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:3...	-9,1kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:3...	-9,1kcal/mol
ΔG = -8,9kcal/mol	2007-06-25 13:3...	-8,9kcal/mol
ΔG = -8,6kcal/mol	2007-06-25 13:3...	-8,6kcal/mol
ΔG = -8,4kcal/mol	2007-06-25 13:3...	-8,4kcal/mol
ΔG = -8,3kcal/mol	2007-06-25 13:3...	-8,3kcal/mol
ΔG = -8,2kcal/mol	2007-06-25 13:3...	-8,2kcal/mol

Elements of structure 'ΔG = -9,9kcal/mol': ΔG = -9,9kcal/mol

- Stem with hairpin at 1..70: ΔG = -8,8kcal/mol
- Dangling nucleotide at 71, dangling from position 70: ΔG =›l

Figure 24.16: *The secondary structure table with the list of structures to the left, and to the right the substructures of the selected structure.*

contributes a free energy given by the sum of its nested substructure energies and energies of its nested structure elements.

The substructure elements to the right are ordered after their occurrence in the sequence; they are described by a region (the sequence positions covered by this substructure) and an energy contribution. Three examples of mixed substructure elements are "Stem base pairs", "Stem with bifurcation" and "Stem with hairpin".

The "Stem base pairs"-substructure is simply a union of stacking elements. It is given by a joined set of base pair positions and an energy contribution displaying the sum of all stacking element-energies.

The "Stem with bifurcation"-substructure defines a substructure enclosed by a specified base pair with and with energy contribution ΔG. The substructure contains a "Stem base pairs"-substructure and a nested bifurcated substructure (multi loop). Also bulge and interior loops can occur separating stem regions.

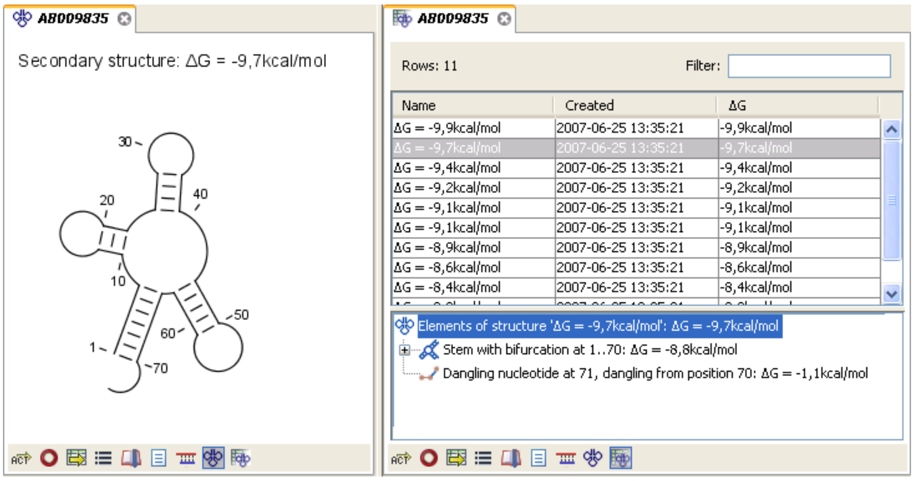
The "Stem with hairpin"-substructure defines a substructure starting at a specified base pair with an enclosed substructure-energy given by ΔG. The substructure contains a "Stem base pairs"-substructure and a hairpin loop. Also bulge and interior loops can occur, separating stem regions.

In order to describe the tree ordering of different substructures, we use an example as a starting point (see figure 24.17).

The structure is a (disjoint) nested union of a "Stem with bifurcation"-substructure and a dangling nucleotide. The nested substructure energies add up to the total energy. The "Stem with bifurcation"-substructure is again a (disjoint) union of a "Stem base pairs"-substructure joining position *1-7* with *64-70* and a multi loop structure element opened at base pair*(7,64)*. To see these structure elements, simply expand the "Stem with bifurcation" node (see figure 24.18).

The multi loop structure element is a union of three "Stem with hairpin"-substructures and contributions to the multi loop opening considering multi loop base pairs and multi loop arcs.

Selecting an element in the table to the right will make a corresponding selection in the **Show Secondary Structure 2D View** (✿) if this is also open and if the "Follow structure selection" has been set in the editors side panel. In figure 24.18 the "Stem with bifurcation" is selected in the



AB009835

Secondary structure: ΔG = -9,7kcal/mol

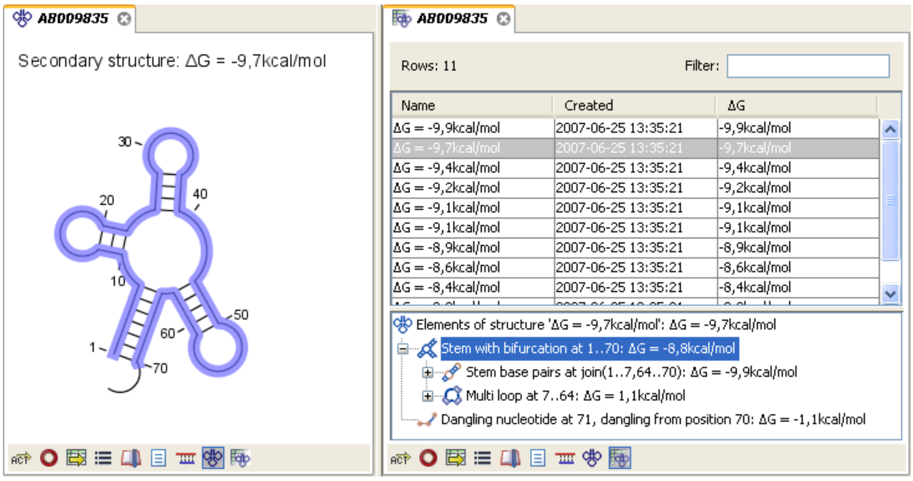
AB009835

Rows: 11 Filter:

Name	Created	ΔG
ΔG = -9,9kcal/mol	2007-06-25 13:35:21	-9,9kcal/mol
ΔG = -9,7kcal/mol	2007-06-25 13:35:21	-9,7kcal/mol
ΔG = -9,4kcal/mol	2007-06-25 13:35:21	-9,4kcal/mol
ΔG = -9,2kcal/mol	2007-06-25 13:35:21	-9,2kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:35:21	-9,1kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:35:21	-9,1kcal/mol
ΔG = -8,9kcal/mol	2007-06-25 13:35:21	-8,9kcal/mol
ΔG = -8,6kcal/mol	2007-06-25 13:35:21	-8,6kcal/mol
ΔG = -8,4kcal/mol	2007-06-25 13:35:21	-8,4kcal/mol

Elements of structure 'ΔG = -9,7kcal/mol': ΔG = -9,7kcal/mol
Stem with bifurcation at 1..70: ΔG = -8,8kcal/mol
Dangling nucleotide at 71, dangling from position 70: ΔG = -1,1kcal/mol

Figure 24.17: *A split view showing a structure table to the right and the secondary structure 2D view to the left.*



AB009835

Secondary structure: ΔG = -9,7kcal/mol

AB009835

Rows: 11 Filter:

Name	Created	ΔG
ΔG = -9,9kcal/mol	2007-06-25 13:35:21	-9,9kcal/mol
ΔG = -9,7kcal/mol	2007-06-25 13:35:21	-9,7kcal/mol
ΔG = -9,4kcal/mol	2007-06-25 13:35:21	-9,4kcal/mol
ΔG = -9,2kcal/mol	2007-06-25 13:35:21	-9,2kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:35:21	-9,1kcal/mol
ΔG = -9,1kcal/mol	2007-06-25 13:35:21	-9,1kcal/mol
ΔG = -8,9kcal/mol	2007-06-25 13:35:21	-8,9kcal/mol
ΔG = -8,6kcal/mol	2007-06-25 13:35:21	-8,6kcal/mol
ΔG = -8,4kcal/mol	2007-06-25 13:35:21	-8,4kcal/mol

Elements of structure 'ΔG = -9,7kcal/mol': ΔG = -9,7kcal/mol
Stem with bifurcation at 1..70: ΔG = -8,8kcal/mol
Stem base pairs at join(1..7,64..70): ΔG = -9,9kcal/mol
Multi loop at 7..64: ΔG = 1,1kcal/mol
Dangling nucleotide at 71, dangling from position 70: ΔG = -1,1kcal/mol

Figure 24.18: *Now the "Stem with bifurcation" node has been selected in the table and a corresponding selection has been made in the view of the secondary structure to the left.*

table, and this part of the structure is high-lighted in the Secondary Structure 2D view.

The correspondence between the table and the structure editor makes it easy to inspect the thermodynamic details of the structure while keeping a visual overview as shown in the above figures.

Handling multiple structures

The table to the left offers a number of tools for working with structures. Select a structure, right-click, and the following menu items will be available:

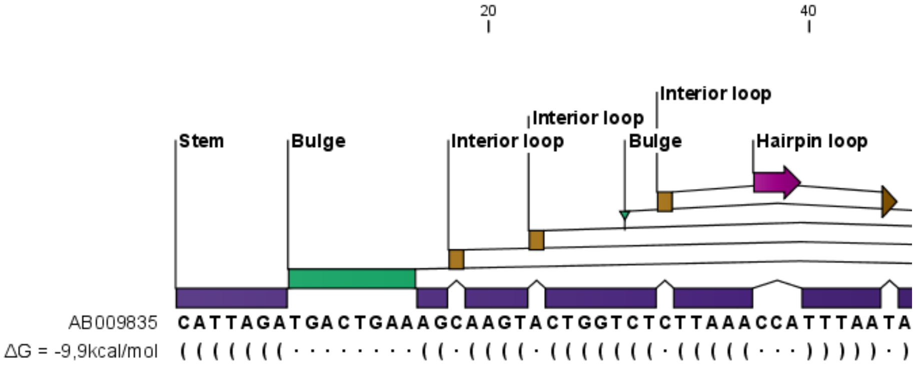
- **Open Secondary Structure in 2D View ( ).** This will open the selected structure in the Secondary structure 2D view.
- **Annotate Sequence with Secondary Structure.** This will add the structure elements as annotations to the sequence. Note that existing structure annotations will be removed.
- **Rename Secondary Structure.** This will allow you to specify a name for the structure to be

displayed in the table.

- **Delete Secondary Structure.** This will delete the selected structure.
- **Delete All Secondary Structures.** This will delete all the selected structures. Note that once you save and close the view, this operation is irreversible. As long as the view is open, you can **Undo** ( ) the operation.

24.2.3 Symbolic representation in sequence view

In the **Side Panel** of normal sequence views ( ), you will find an extra group under **Nucleotide info** called **Secondary Structure**. This is used to display a symbolic representation of the secondary structure along the sequence (see figure 24.19).



```
AB009835          CATTAGATGACTGAAAGCAAGTACTGGTCTCTTAAACCATTTAATA
ΔG = -9,9kcal/mol (((((((........((.((((.(((((((.(((((...))))).)
```

Figure 24.19: *The secondary structure visualized below the sequence and with annotations shown above.*

The following options can be set:

- **Show all structures.** If more than one structure is predicted, this option can be used if all the structures should be displayed.
- **Show first.** If not all structures are shown, this can be used to determine the number of structures to be shown.
- **Sort by.** When you select to display e.g. four out of eight structures, this option determines which the "first four" should be.
  - Sort by ΔG.
  - Sort by name.
  - Sort by time of creation.

  If these three options do not provide enough control, you can rename the structures in a meaningful alphabetical way so that you can use the "name" to display the desired ones.
- **Match symbols.** How a base pair should be represented.
- **No match symbol.** How bases which are not part of a base pair should be represented.

- **Height.** When you zoom out, this option determines the height of the symbols as shown in figure 24.20 (when zoomed in, there is no need for specifying the height).
- **Base pair probability.** See section 24.2.4 below).

When you zoom in and out, the appearance of the symbols change. In figure 24.19, the view is zoomed in. In figure 24.20 you see the same sequence zoomed out to fit the width of the sequence.

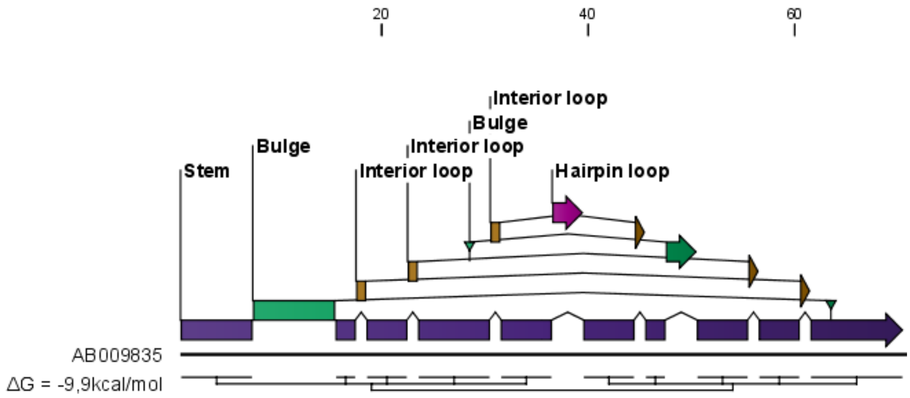


Figure 24.20: *The secondary structure visualized below the sequence and with annotations shown above. The view is zoomed out to fit the width of the sequence.*

24.2.4 Probability-based coloring

In the **Side Panel** of both linear and secondary structure 2D views, you can choose to color structure symbols and sequence residues according to the probability of base pairing / not base pairing, as shown in figure 24.4.

In the linear sequence view ( ), this is found in **Nucleotide info** under **Secondary structure**, and in the secondary structure 2D view ( ), it is found under **Residue coloring**.

For both paired and unpaired bases, you can set the foreground color and the background color to a gradient with the color at the left side indicating a probability of 0, and the color at the right side indicating a probability of 1.

Note that you have to **Zoom to 100%** ( ) in order to see the coloring.

24.3 Evaluate structure hypothesis

Hypotheses about an RNA structure can be tested using *CLC Genomics Workbench*. A structure hypothesis *H* is formulated using the structural constraint annotations described in section 24.1.4. By adding several annotations complex structural hypotheses can be formulated (see 24.21).

Given the set *S* of all possible structures, only a subset of these S_H will comply with the formulated hypotheses. We can now find the probability of *H* as:

$$P(H) = \frac{\sum\limits_{s_H \in S_H} P(s_H)}{\sum\limits_{s \in S} P(s)} = \frac{PF_H}{PF_{\text{full}}},$$

where PF_H is the partition function calculated for all structures permissible by H (S_H) and PF_{full} is the full partition function. Calculating the probability can thus be done with two passes of the partition function calculation, one with structural constraints, and one without. 24.21.

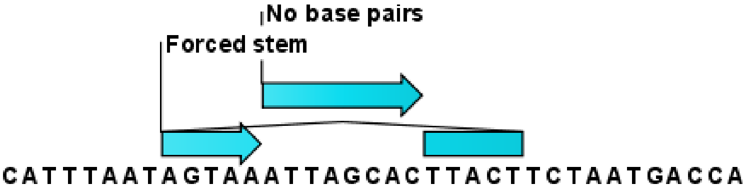


Figure 24.21: *Two constraints defining a structural hypothesis.*

24.3.1 Selecting sequences for evaluation

The evaluation is started from the **Toolbox**:

Toolbox | **RNA Structure** ( ) | **Evaluate Structure Hypothesis** ( )

This opens the dialog shown in figure 24.22.

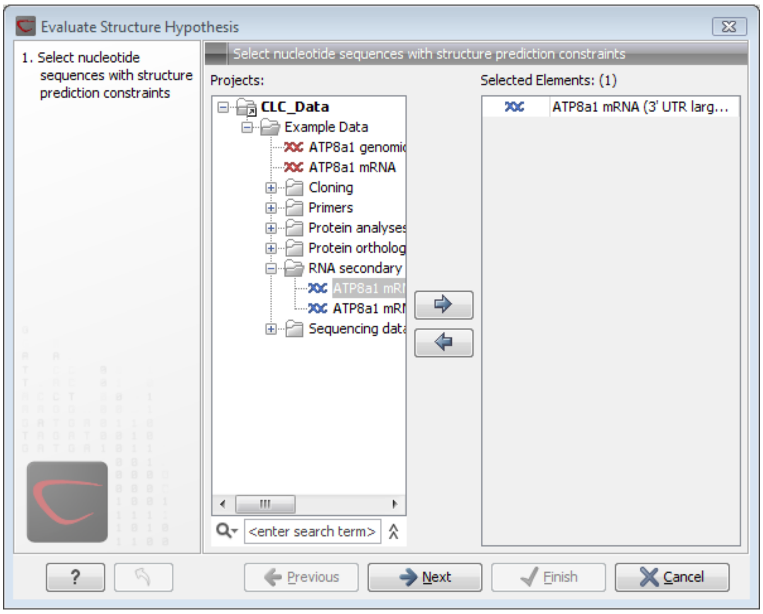


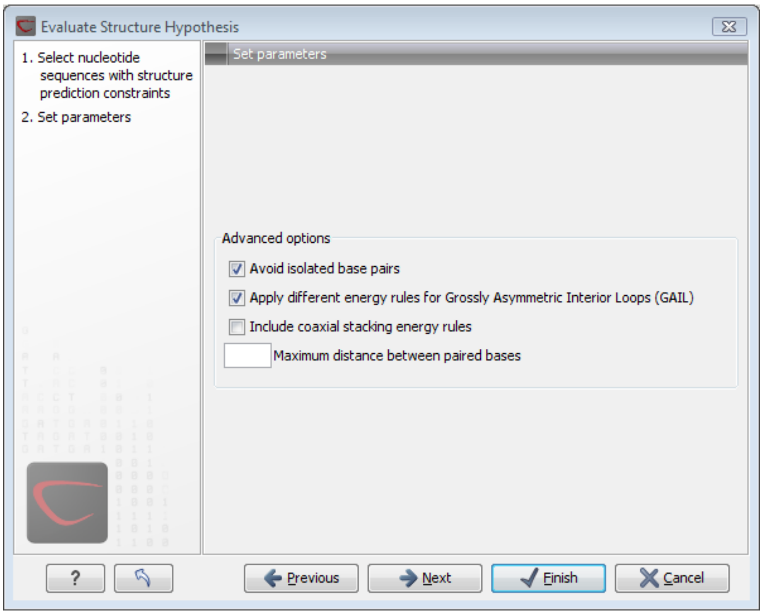
Figure 24.22: *Selecting RNA or DNA sequences for evaluating structure hypothesis.*

If you have selected sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements. Note, that the selected sequences must contain a structure hypothesis in the form of manually added constraint annotations.

Click **Next** to adjust evaluation parameters (see figure 24.23).

The partition function algorithm includes a number of advanced options:

- **Avoid isolated base pairs**. The algorithm filters out isolated base pairs (i.e. stems of length 1).
- **Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL)**. Compute the minimum free energy applying different rules for Grossly Asymmetry Interior Loops (GAIL). A Grossly Asymmetry Interior Loop (GAIL) is an interior loop that is $1 \times n$ or $n \times 1$ where $n > 2$ (see http://www.bioinfo.rpi.edu/~zukerm/lectures/RNAfold-html/rnafold-print.pdf).
- **Include coaxial stacking energy rules**. Include free energy increments of coaxial stacking for adjacent helices [Mathews et al., 2004].


Evaluate Structure Hypothesis
1. Select nucleotide sequences with structure prediction constraints
2. Set parameters
Set parameters
? Previous Next Finish Cancel

Advanced options

☑ Avoid isolated base pairs

☑ Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL)

☐ Include coaxial stacking energy rules

Maximum distance between paired bases

Figure 24.23: *Adjusting parameters for hypothesis evaluation.*

24.3.2 Probabilities

After evaluation of the structure hypothesis an annotation is added to the input sequence. This annotation covers the same region as the annotations that constituted the hypothesis and contains information about the probability of the evaluated hypothesis (see figure 24.24).

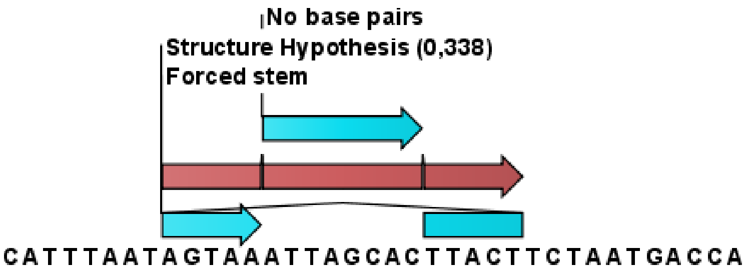

No base pairs
Structure Hypothesis (0,338)
Forced stem
CATTTAATAGTAAATTAGCACTTACTTCTAATGACCA

Figure 24.24: *This hypothesis has a probability of 0.338 as shown in the annotation.*

24.4 Structure Scanning Plot

In *CLC Genomics Workbench* it is possible to scan larger sequences for the existence of local conserved RNA structures. The structure scanning approach is similar in spirit to the works of [Workman and Krogh, 1999] and [Clote et al., 2005]. The idea is that if natural selection is operating to maintain a stable local structure in a given region, then the minimum free energy of the region will be markedly lower than the minimum free energy found when the nucleotides of the subsequence are distributed in random order.

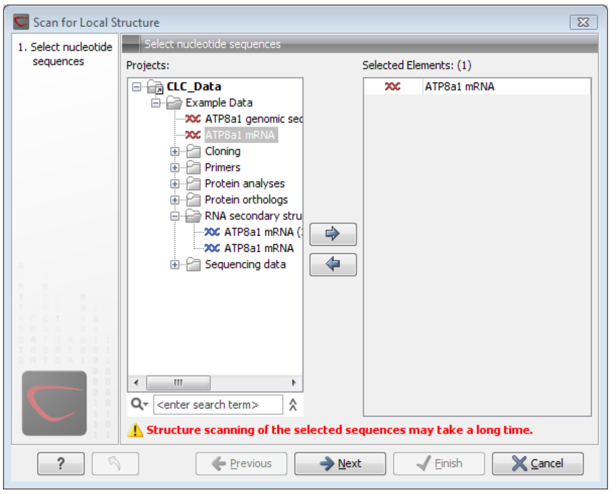
The algorithm works by sliding a window along the sequence. Within the window, the minimum free energy of the subsequence is calculated. To evaluate the significance of the local structure signal its minimum free energy is compared to a background distribution of minimum free energies obtained from shuffled sequences, using Z-scores [Rivas and Eddy, 2000]. The Z-score statistics corresponds to the number of standard deviations by which the minimum free energy of the original sequence deviates from the average energy of the shuffled sequences. For a given Z-score, the statistical significance is evaluated as the probability of observing a more extreme Z-score under the assumption that Z-scores are normally distributed [Rivas and Eddy, 2000].

24.4.1 Selecting sequences for scanning

The scanning is started from the **Toolbox**:

Toolbox | RNA Structure ( ) | Evaluate Structure Hypothesis ( )

This opens the dialog shown in figure 24.25.



Structure scanning of the selected sequences may take a long time.

Figure 24.25: *Selecting RNA or DNA sequences for structure scanning.*

If you have selected sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** to adjust scanning parameters (see figure 24.26).

The first group of parameters pertain to the methods of sequence resampling. There are four ways of resampling, all described in detail in [Clote et al., 2005]:

- **Mononucleotide shuffling.** Shuffle method generating a sequence of the exact same mononucleotide frequency

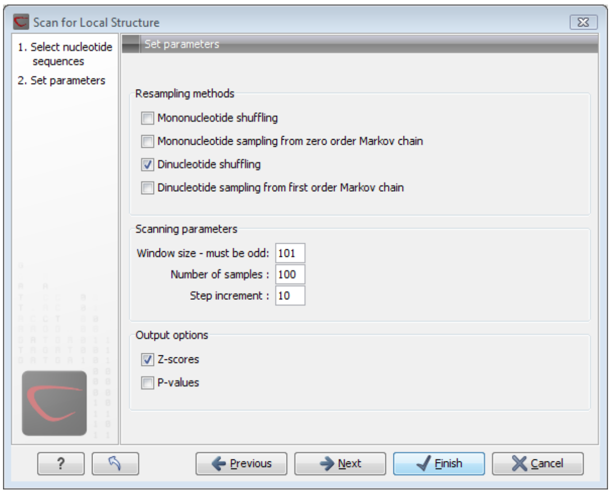
- **Dinucleotide shuffling.** Shuffle method generating a sequence of the exact same dinucleotide frequency
- **Mononucleotide sampling from zero order Markov chain.** Resampling method generating a sequence of the same expected mononucleotide frequency.
- **Dinucleotide sampling from first order Markov chain.** Resampling method generating a sequence of the same expected dinucleotide frequency.

The second group of parameters pertain to the scanning settings and include:

- **Window size.** The width of the sliding window.
- **Number of samples.** The number of times the sequence is resampled to produce the background distribution.
- **Step increment.** Step increment when plotting sequence positions against scoring values.

The third parameter group contains the output options:

- **Z-scores.** Create a plot of Z-scores as a function of sequence position.
- **P-values.** Create a plot of the statistical significance of the structure signal as a function of sequence position.



Resampling methods

□ Mononucleotide shuffling

□ Mononucleotide sampling from zero order Markov chain

☑ Dinucleotide shuffling

□ Dinucleotide sampling from first order Markov chain

Scanning parameters

Window size - must be odd: 101

Number of samples : 100

Step increment : 10

Output options

☑ Z-scores

□ P-values

Figure 24.26: *Adjusting parameters for structure scanning.*

24.4.2 The structure scanning result

The output of the analysis are plots of Z-scores and probabilities as a function of sequence position. A strong propensity for local structure can be seen as spikes in the graphs (see figure 24.27).

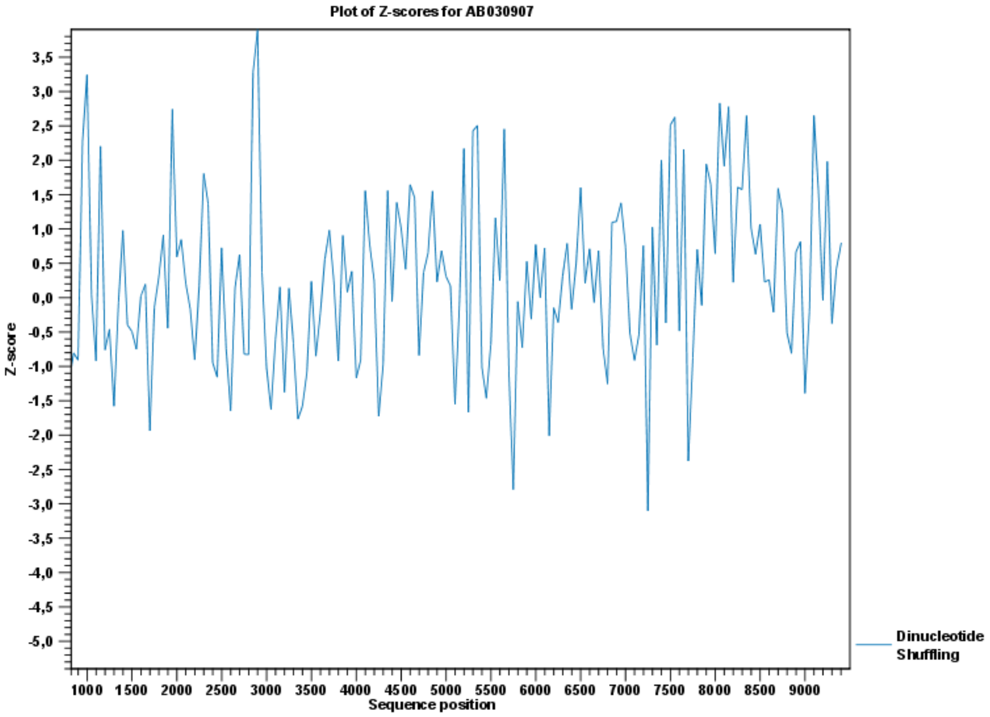


Figure 24.27: *A plot of the Z-scores produced by sliding a window along a sequence.*

24.5 Bioinformatics explained: RNA structure prediction by minimum free energy minimization

RNA molecules are hugely important in the biology of the cell. Besides their rather simple role as an intermediate messenger between DNA and protein, RNA molecules can have a plethora of biologic functions. Well known examples of this are the infrastructural RNAs such as tRNAs,rRNAs and snRNAs, but the existence and functionality of several other groups of non-coding RNAs are currently being discovered. These include micro- (miRNA), small interfering- (siRNA), Piwi interacting- (piRNA) and small modulatory RNAs (smRNA) [Costa, 2007].

A common feature of many of these non-coding RNAs is that the molecular structure is important for the biological function of the molecule.

Ideally, biological function is best interpreted against a 3D structure of an RNA molecule. However, 3D structure determination of RNA molecules is time-consuming, expensive, and difficult [Shapiro et al., 2007] and there is therefore a great disparity between the number of known RNA sequences and the number of known RNA 3D structures.

However, as it is the case for proteins, RNA tertiary structures can be characterized by secondary structural elements. These are defined by hydrogen bonds within the molecule that form several recognizable "domains" of secondary structure like stems, hairpin loops, bulges and internal loops (see below). Furthermore, the high degree of base-pair conservation observed in the evolution of RNA molecules shows that a large part of the functional information is actually contained in the secondary structure of the RNA molecule.

Fortunately, RNA secondary structure can be computationally predicted from sequence data allowing researchers to map sequence information to functional information. The subject of this paper is to describe a very popular way of doing this, namely free energy minimization. For an in-depth review of algorithmic details, we refer the reader to [Mathews and Turner, 2006].

24.5.1 The algorithm

Consider an RNA molecule and one of its possible structures S_1. In a stable solution there will be an equilibrium between unstructured RNA strands and RNA strands folded into S_1. The propensity of a strand to leave a structure such as S_1 (the stability of S_1), is determined by the free energy change involved in its formation. The structure with the lowest free energy (S_{min}) is the most stable and will also be the most represented structure at equilibrium. The objective of minimum free energy (MFE) folding is therefore to identify S_{min} amongst all possible structures.

In the following, we only consider structures without pseudoknots, i.e. structures that do not contain any non-nested base pairs.

Under this assumption, a sequence can be folded into a single coherent structure or several sequential structures that are joined by unstructured regions. Each of these structures is a union of well described structure elements (see below for a description of these). The free energy for a given structure is calculated by an additive nearest neighbor model. Additive, means that the total free energy of a secondary structure is the sum of the free energies of its individual structural elements. Nearest neighbor, means that the free energy of each structure element depends only on the residues it contains and on the most adjacent Watson-Crick base pairs.

The simplest method to identify S_{min} would be to explicitly generate all possible structures, but it can be shown that the number of possible structures for a sequence grows exponentially with the sequence length [Zuker and Sankoff, 1984] leaving this approach unfeasible. Fortunately, a two step algorithm can be constructed which implicitly surveys all possible structures without explicitly generating the structures [Zuker and Stiegler, 1981]: The first step determines the free energy for each possible sequence fragment starting with the shortest fragments. Here, the lowest free energy for longer fragments can be expediently calculated from the free energies of the smaller sub-sequences they contain. When this process reaches the longest fragment, i.e., the complete sequence, the MFE of the entire molecule is known. The second step is called traceback, and uses all the free energies computed in the first step to determine S_{min} - the exact structure associated with the MFE. Acceptable calculation speed is achieved by using *dynamic programming* where sub-sequence results are saved to avoid recalculation. However, this comes at the price of a higher requirement for computer memory.

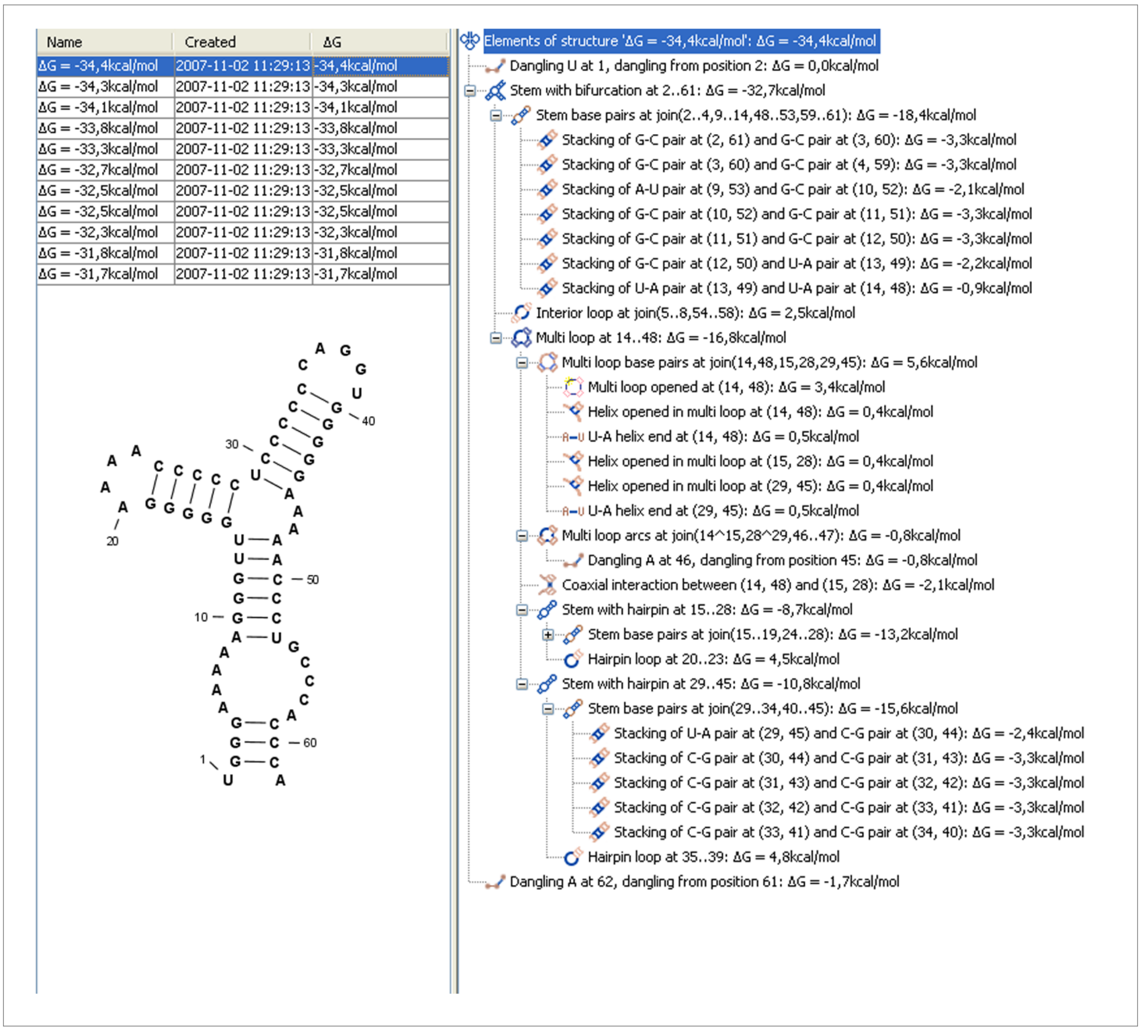
The structure element energies that are used in the recursions of these two steps, are derived from empirical calorimetric experiments performed on small molecules see e.g. [Mathews et al., 1999].

Suboptimal structures determination

A number of known factors violate the assumptions that are implicit in MFE structure prediction. [Schroeder et al., 1999] and [Chen et al., 2004] have shown experimental indications that the thermodynamic parameters are sequence dependent. Moreover, [Longfellow et al., 1990] and [Kierzek et al., 1999], have demonstrated that some structural elements show non-nearest neighbor effects. Finally, single stranded nucleotides in multi loops are known to influence stability [Mathews and Turner, 2002].

These phenomena can be expected to limit the accuracy of RNA secondary structure prediction by free energy minimization and it should be clear that the predicted MFE structure may deviate somewhat from the actual preferred structure of the molecule. This means that it may be informative to inspect the landscape of suboptimal structures which surround the MFE structure to look for general structural properties which seem to be robust to minor variations in the total free energy of the structure.

An effective procedure for generating a sample of suboptimal structures is given in [Zuker, 1989a]. This algorithm works by going through all possible Watson-Crick base pair in the molecule. For each of these base pairs, the algorithm computes the most optimal structure among all the structures that contain this pair, see figure 24.28.



Name	Created	ΔG
ΔG = -34,4kcal/mol	2007-11-02 11:29:13	-34,4kcal/mol
ΔG = -34,3kcal/mol	2007-11-02 11:29:13	-34,3kcal/mol
ΔG = -34,1kcal/mol	2007-11-02 11:29:13	-34,1kcal/mol
ΔG = -33,8kcal/mol	2007-11-02 11:29:13	-33,8kcal/mol
ΔG = -33,3kcal/mol	2007-11-02 11:29:13	-33,3kcal/mol
ΔG = -32,7kcal/mol	2007-11-02 11:29:13	-32,7kcal/mol
ΔG = -32,5kcal/mol	2007-11-02 11:29:13	-32,5kcal/mol
ΔG = -32,5kcal/mol	2007-11-02 11:29:13	-32,5kcal/mol
ΔG = -32,3kcal/mol	2007-11-02 11:29:13	-32,3kcal/mol
ΔG = -31,8kcal/mol	2007-11-02 11:29:13	-31,8kcal/mol
ΔG = -31,7kcal/mol	2007-11-02 11:29:13	-31,7kcal/mol

```
Elements of structure 'ΔG = -34,4kcal/mol': ΔG = -34,4kcal/mol
  Dangling U at 1, dangling from position 2: ΔG = 0,0kcal/mol
  Stem with bifurcation at 2..61: ΔG = -32,7kcal/mol
    Stem base pairs at join(2..4,9..14,48..53,59..61): ΔG = -18,4kcal/mol
      Stacking of G-C pair at (2, 61) and G-C pair at (3, 60): ΔG = -3,3kcal/mol
      Stacking of G-C pair at (3, 60) and G-C pair at (4, 59): ΔG = -3,3kcal/mol
      Stacking of A-U pair at (9, 53) and G-C pair at (10, 52): ΔG = -2,1kcal/mol
      Stacking of G-C pair at (10, 52) and G-C pair at (11, 51): ΔG = -3,3kcal/mol
      Stacking of G-C pair at (11, 51) and G-C pair at (12, 50): ΔG = -3,3kcal/mol
      Stacking of G-C pair at (12, 50) and U-A pair at (13, 49): ΔG = -2,2kcal/mol
      Stacking of U-A pair at (13, 49) and U-A pair at (14, 48): ΔG = -0,9kcal/mol
    Interior loop at join(5..8,54..58): ΔG = 2,5kcal/mol
    Multi loop at 14..48: ΔG = -16,8kcal/mol
      Multi loop base pairs at join(14,48,15,28,29,45): ΔG = 5,6kcal/mol
        Multi loop opened at (14, 48): ΔG = 3,4kcal/mol
        Helix opened in multi loop at (14, 48): ΔG = 0,4kcal/mol
        A–U U-A helix end at (14, 48): ΔG = 0,5kcal/mol
        Helix opened in multi loop at (15, 28): ΔG = 0,4kcal/mol
        Helix opened in multi loop at (29, 45): ΔG = 0,4kcal/mol
        A–U U-A helix end at (29, 45): ΔG = 0,5kcal/mol
      Multi loop arcs at join(14^15,28^29,46..47): ΔG = -0,8kcal/mol
        Dangling A at 46, dangling from position 45: ΔG = -0,8kcal/mol
      Coaxial interaction between (14, 48) and (15, 28): ΔG = -2,1kcal/mol
      Stem with hairpin at 15..28: ΔG = -8,7kcal/mol
        Stem base pairs at join(15..19,24..28): ΔG = -13,2kcal/mol
        Hairpin loop at 20..23: ΔG = 4,5kcal/mol
      Stem with hairpin at 29..45: ΔG = -10,8kcal/mol
        Stem base pairs at join(29..34,40..45): ΔG = -15,6kcal/mol
          Stacking of U-A pair at (29, 45) and C-G pair at (30, 44): ΔG = -2,4kcal/mol
          Stacking of C-G pair at (30, 44) and C-G pair at (31, 43): ΔG = -3,3kcal/mol
          Stacking of C-G pair at (31, 43) and C-G pair at (32, 42): ΔG = -3,3kcal/mol
          Stacking of C-G pair at (32, 42) and C-G pair at (33, 41): ΔG = -3,3kcal/mol
          Stacking of C-G pair at (33, 41) and C-G pair at (34, 40): ΔG = -3,3kcal/mol
        Hairpin loop at 35..39: ΔG = 4,8kcal/mol
  Dangling A at 62, dangling from position 61: ΔG = -1,7kcal/mol
```

Figure 24.28: *A number of suboptimal structures have been predicted using* **CLC Genomics Workbench** *and are listed at the top left. At the right hand side, the structural components of the selected structure are listed in a hierarchical structure and on the left hand side the structure is displayed.*

24.5.2 Structure elements and their energy contribution

In this section, we classify the structure elements defining a secondary structure and describe their energy contribution.

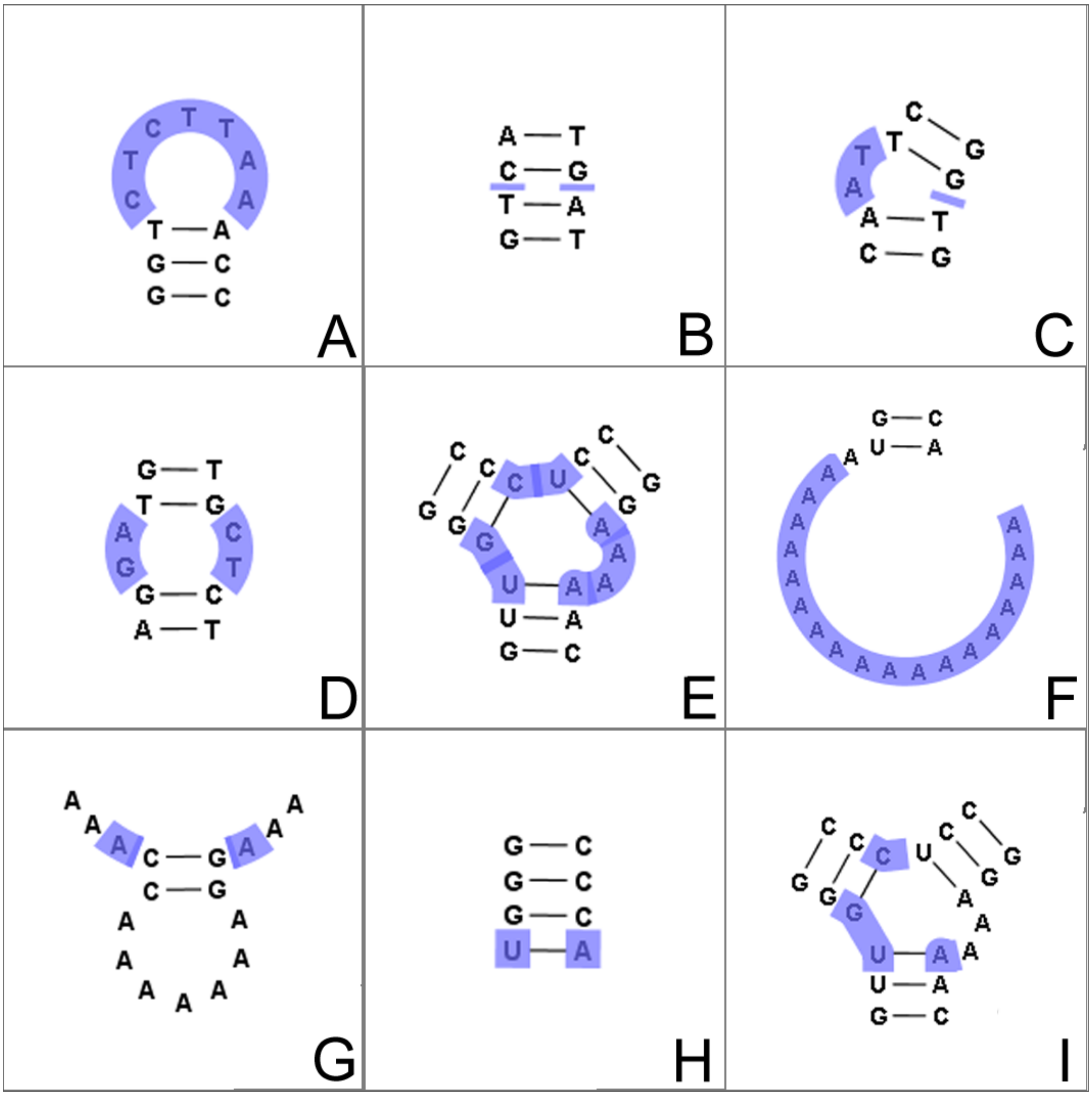


Figure 24.29: *The different structure elements of RNA secondary structures predicted with the free energy minimization algorithm in* **CLC Genomics Workbench**. *See text for a detailed description.*

Nested structure elements

The structure elements involving nested base pairs can be classified by a given base pair and the other base pairs that are nested and *accessible* from this pair. For a more elaborate description we refer the reader to [Sankoff et al., 1983] and [Zuker and Sankoff, 1984].

If the nucleotides with position number (i, j) form a base pair and $i < k, l < j$, then we say that the base pair (k, l) is **accessible** from (i, j) if there is no intermediate base pair (i', j') such that $i < i' < k, l < j' < j$. This means that (k, l) is nested within the pair i, j and there is no other base pair in between.

Using the number of accessible pase pairs, we can define the following distinct structure elements:

1. **Hairpin loop ().** A base pair with 0 other accessible base pairs forms a *hairpin loop*. The energy contribution of a hairpin is determined by the length of the unpaired (loop) region and the two bases adjacent to the closing base pair which is termed a terminal mismatch (see figure 24.29A).
2. A base pair with 1 accessible base pair can give rise to three distinct structure elements:
   - **Stacking of base pairs ().** A *stacking* of two consecutive pairs occur if $i' - i = 1 = j - j'$. Only canonical base pairs ($A - U$ or $G - C$ or $G - U$) are allowed (see figure 24.29B). The energy contribution is determined by the type and order of the two base pairs.
   - **Bulge ().** A *bulge loop* occurs if $i' - i > 1$ or $j - j' > 1$, but not both. This means that the two base pairs enclose an unpaired region of length 0 on one side and an unpaired region of length ≥ 1 on the other side (see figure 24.29C). The energy contribution of a bulge is determined by the length of the unpaired (loop) region and the two closing base pairs.
   - **Interior loop ().**An *interior loop* occurs if both $i' - i > 1$ and $i - j' > 1$ This means that the two base pairs enclose an unpaired region of length ≥ 1 on both sides (see figure 24.29D). The energy contribution of an interior loop is determined by the length of the unpaired (loop) region and the four unpaired bases adjacent to the opening- and the closing base pair.
3. **Multi loop opened ().** A base pair with more than two accessible base pairs gives rise to a *multi loop*, a loop from which three or more stems are opened (see figure 24.29E). The energy contribution of a multi loop depends on the number of **Stems opened in multi-loop ()** that protrude from the loop.

Other structure elements

- A collection of single stranded bases not accessible from any base pair is called an *exterior (or external) loop* (see figure 24.29F). These regions do not contribute to the total free energy.
- **Dangling nucleotide ().** A *dangling nucleotide* is a single stranded nucleotide that forms a stacking interaction with an adjacent base pair. A dangling nucleotide can be a $3'$ or $5'$-dangling nucleotide depending on the orientation (see figure 24.29G). The energy contribution is determined by the single stranded nucleotide, its orientation and on the adjacent base pair.
- **Non-GC terminating stem (A–U).** If a base pair other than a G-C pair is found at the end of a stem, an energy penalty is assigned (see figure 24.29H).
- **Coaxial interaction ().** Coaxial stacking is a favorable interaction of two stems where the base pairs at the ends can form a stacking interaction. This can occur between stems in a multi loop and between the stems of two different sequential structures. Coaxial stacking can occur between stems with no intervening nucleotides (adjacent stems) and between stems with one intervening nucleotide from each strand (see figure 24.29I). The energy contribution is determined by the adjacent base pairs and the intervening nucleotides.

Experimental constraints

A number of techniques are available for probing RNA structures. These techniques can determine individual components of an existing structure such as the existence of a given base pair. It is possible to add such experimental constraints to the secondary structure prediction based on free energy minimization (see figure 24.30) and it has been shown that this can dramatically increase the fidelity of the secondary structure prediction [Mathews and Turner, 2006].

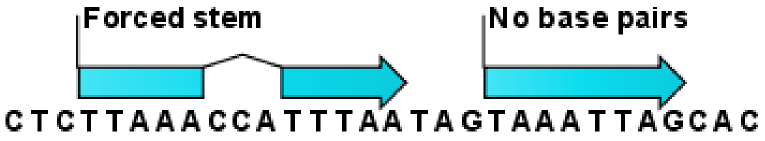


Figure 24.30: *Known structural features can be added as constraints to the secondary structure prediction algorithm in* **CLC Genomics Workbench***.*

Creative Commons License

Part IV

Appendix

Appendix A

Comparison of workbenches

Below we list a number of functionalities that differ between CLC Workbenches and the CLC Sequence Viewer:

- CLC Sequence Viewer (🟥)
- CLC Protein Workbench (🟩)
- CLC DNA Workbench (🟦)
- CLC RNA Workbench (🟪)
- CLC Main Workbench (🟨)
- CLC Genomics Workbench (🟦)

Data handling	Viewer	Protein	DNA	RNA	Main	Genomics
Add multiple locations to Navigation Area		🟩	🟦	🟪	🟨	🟦
Share data on network drive		🟩	🟦	🟪	🟨	🟦
Search all your data		🟩	🟦	🟪	🟨	🟦

Assembly of sequencing data	Viewer	Protein	DNA	RNA	Main	Genomics
Advanced contig assembly			🟦		🟨	🟦
Importing and viewing trace data			🟦		🟨	🟦
Trim sequences			🟦		🟨	🟦
Assemble without use of reference sequence			🟦		🟨	🟦
Map to reference sequence			🟦		🟨	🟦
Assemble to existing contig			🟦		🟨	🟦
Viewing and edit contigs			🟦		🟨	🟦
Tabular view of an assembled contig (easy data overview)			🟦		🟨	🟦
Secondary peak calling			🟦		🟨	🟦
Multiplexing based on barcode or name			🟦		🟨	🟦

Next-generation Sequencing Data Analysis	Viewer	Protein	DNA	RNA	Main	Genomics
Import of 454, Illumina Genome Analyzer, SOLiD and Helicos data						🟦
Reference assembly of human-size genomes						🟦
De novo assembly						🟦
SNP/DIP detection						🟦
Graphical display of large contigs						🟦
Support for mixed-data assembly						🟦
Paired data support						🟦
RNA-Seq analysis						🟦
Expression profiling by tags						🟦
ChIP-Seq analysis						🟦

Expression Analysis	Viewer	Protein	DNA	RNA	Main	Genomics
Import of Illumina BeadChip, Affymetrix, GEO data					🟨	🟦
Import of Gene Ontology annotation files					🟨	🟦
Import of Custom expression data table and Custom annotation files					🟨	🟦
Multigroup comparisons					🟨	🟦
Advanced plots: scatter plot, volcano plot, box plot and MA plot					🟨	🟦
Hierarchical clustering					🟨	🟦
Statistical analysis on count-based and gaussian data					🟨	🟦
Annotation tests					🟨	🟦
Principal component analysis (PCA)					🟨	🟦
Hierarchical clustering and heat maps					🟨	🟦
Analysis of RNA-Seq/Tag profiling samples					🟨	🟦

Molecular cloning	Viewer	Protein	DNA	RNA	Main	Genomics
Advanced molecular cloning			🟦		🟨	🟦
Graphical display of in silico cloning			🟦		🟨	🟦
Advanced sequence manipulation			🟦		🟨	🟦

Database searches	Viewer	Protein	DNA	RNA	Main	Genomics
GenBank Entrez searches	🟥	🟩	🟦	🟪	🟨	🟦
UniProt searches (Swiss-Prot/TrEMBL)		🟩			🟨	🟦
Web-based sequence search using BLAST		🟩	🟦	🟪	🟨	🟦
BLAST on local database		🟩	🟦	🟪	🟨	🟦
Creation of local BLAST database		🟩	🟦	🟪	🟨	🟦
PubMed lookup		🟩	🟦	🟪	🟨	🟦
Web-based lookup of sequence data		🟩	🟦	🟪	🟨	🟦
Search for structures (at NCBI)		🟩			🟨	🟦

General sequence analyses	Viewer	Protein	DNA	RNA	Main	Genomics
Linear sequence view	🟥	🟩	🟦	🟪	🟨	🟦
Circular sequence view	🟥	🟩	🟦	🟪	🟨	🟦
Text based sequence view		🟩	🟦	🟪	🟨	🟦
Editing sequences	🟥	🟩	🟦	🟪	🟨	🟦
Adding and editing sequence annotations		🟩	🟦	🟪	🟨	🟦
Advanced annotation table		🟩	🟦	🟪	🟨	🟦
Join multiple sequences into one	🟥	🟩	🟦	🟪	🟨	🟦
Sequence statistics	🟥	🟩	🟦	🟪	🟨	🟦
Shuffle sequence	🟥	🟩	🟦	🟪	🟨	🟦
Local complexity region analyses		🟩	🟦	🟪	🟨	🟦
Advanced protein statistics		🟩			🟨	🟦
Comprehensive protein characteristics repor		🟩			🟨	🟦

Nucleotide analyses	Viewer	Protein	DNA	RNA	Main	Genomics
Basic gene finding	🟥	🟩	🟦	🟪	🟨	🟦
Reverse complement without loss of annotation	🟥	🟩	🟦	🟪	🟨	🟦
Restriction site analysis	🟥	🟩	🟦	🟪	🟨	🟦
Advanced interactive restriction site analysis			🟦	🟪	🟨	🟦
Translation of sequences from DNA to proteins	🟥	🟩	🟦	🟪	🟨	🟦
Interactive translations of sequences anc alignments		🟩	🟦	🟪	🟨	🟦
G/C content analyses and graphs		🟩	🟦	🟪	🟨	🟦

Protein analyses	Viewer	Protein	DNA	RNA	Main	Genomics
3D molecule view		🟩			🟨	🟦
Hydrophobicity analyses		🟩	🟦	🟪	🟨	🟦
Antigenicity analysis		🟩			🟨	🟦
Protein charge analysis		🟩			🟨	🟦
Reverse translation from protein to DNA		🟩	🟦	🟪	🟨	🟦
Proteolytic cleavage detection		🟩			🟨	🟦
Prediction of signal peptides (SignalP)		🟩			🟨	🟦
Transmembrane helix prediction (TMHMM)		🟩			🟨	🟦
Secondary protein structure prediction		🟩			🟨	🟦
PFAM domain search		🟩			🟨	🟦

Sequence alignment	Viewer	Protein	DNA	RNA	Main	Genomics
Multiple sequence alignments (Two algorithms)	🟥	🟩	🟦	🟪	🟨	🟦
Advanced re-alignment and fix-point align ment options		🟩	🟦	🟪	🟨	🟦
Advanced alignment editing options	🟥	🟩	🟦	🟪	🟨	🟦
Join multiple alignments into one		🟩	🟦	🟪	🟨	🟦
Consensus sequence determination and management	🟥	🟩	🟦	🟪	🟨	🟦
Conservation score along sequences	🟥	🟩	🟦	🟪	🟨	🟦
Sequence logo graphs along alignments		🟩	🟦	🟪	🟨	🟦
Gap fraction graphs		🟩	🟦	🟪	🟨	🟦
Copy annotations between sequences in alignments		🟩	🟦	🟪	🟨	🟦
Pairwise comparison	🟥	🟩	🟦	🟪	🟨	🟦

RNA secondary structure	Viewer	Protein	DNA	RNA	Main	Genomics
Advanced prediction of RNA secondary structure				🟪	🟨	🟦
Integrated use of base pairing constraints				🟪	🟨	🟦
Graphical view and editing of secondary structure				🟪	🟨	🟦
Info about energy contributions of structure elements				🟪	🟨	🟦
Prediction of multiple sub-optimal structures				🟪	🟨	🟦
Evaluate structure hypothesis				🟪	🟨	🟦
Structure scanning				🟪	🟨	🟦
Partition function				🟪	🟨	🟦

Dot plots	Viewer	Protein	DNA	RNA	Main	Genomics
Dot plot based analyses		🟩	🟦	🟪	🟨	🟦

Phylogenetic trees	Viewer	Protein	DNA	RNA	Main	Genomics
Neighbor-joining and UPGMA phylogenies	🟥	🟩	🟦	🟪	🟨	🟦
Maximum likelihood phylogeny of nucleotides		🟩	🟦	🟪	🟨	🟦

Pattern discovery	Viewer	Protein	DNA	RNA	Main	Genomics
Search for sequence match	🟥	🟩	🟦	🟪	🟨	🟦
Motif search for basic patterns		🟩	🟦	🟪	🟨	🟦
Motif search with regular expressions		🟩	🟦	🟪	🟨	🟦
Motif search with ProSite patterns		🟩	🟦	🟪	🟨	🟦
Pattern discovery		🟩	🟦	🟪	🟨	🟦

Primer design	Viewer	Protein	DNA	RNA	Main	Genomics
Advanced primer design tools			🟦		🟨	🟦
Detailed primer and probe parameters			🟦		🟨	🟦
Graphical display of primers			🟦		🟨	🟦
Generation of primer design output			🟦		🟨	🟦
Support for Standard PCR			🟦		🟨	🟦
Support for Nested PCR			🟦		🟨	🟦
Support for TaqMan PCR			🟦		🟨	🟦
Support for Sequencing primers			🟦		🟨	🟦
Alignment based primer design			🟦		🟨	🟦
Alignment based TaqMan probedesign			🟦		🟨	🟦
Match primer with sequence			🟦		🟨	🟦
Ordering of primers			🟦		🟨	🟦
Advanced analysis of primer properties			🟦		🟨	🟦

Molecular cloning	Viewer	Protein	DNA	RNA	Main	Genomics
Advanced molecular cloning			🟦		🟨	🟦
Graphical display of in silico cloning			🟦		🟨	🟦
Advanced sequence manipulation			🟦		🟨	🟦

SNP annotation using BLAST	Viewer	Protein	DNA	RNA	Main	Genomics
Integrated BLAST at SNP database			🟦		🟨	🟦
Annotate sequence with SNP's			🟦		🟨	🟦

Virtual gel view	Viewer	Protein	DNA	RNA	Main	Genomics
Fully integrated virtual 1D DNA gel simulator			🟦		🟨	🟦

For a more detailed comparison, we refer to `http://www.clcbio.com/compare`.

Appendix B

Graph preferences

This section explains the view settings of graphs. The **Graph preferences** at the top of the **Side Panel** includes the following settings:

- **Lock axes.** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame.** Shows a frame around the graph.
- **Show legends.** Shows the data legends.
- **Tick type**. Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at.** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range**. Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range**. Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **X-axis at zero**. This will draw the x axis at y = 0. Note that the axis range will not be changed.
- **Y-axis at zero**. This will draw the y axis at x = 0. Note that the axis range will not be changed.
- **Show as histogram**. For some data-series it is possible to see the graph as a histogram rather than a line plot.

The **Lines and plots** below contains the following settings:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Line width**
  - Thin
  - Medium
  - Wide
- **Line type**
  - None
  - Line
  - Long dash
  - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

For graphs with multiple data series, you can select which curve the dot and line preferences should apply to. This setting is at the top of the **Side Panel** group.

Note that the graph title and the axes titles can be edited simply by clicking with the mouse. These changes will be saved when you **Save** (💾) the graph - whereas the changes in the **Side Panel** need to be saved explicitly (see section 5.6).

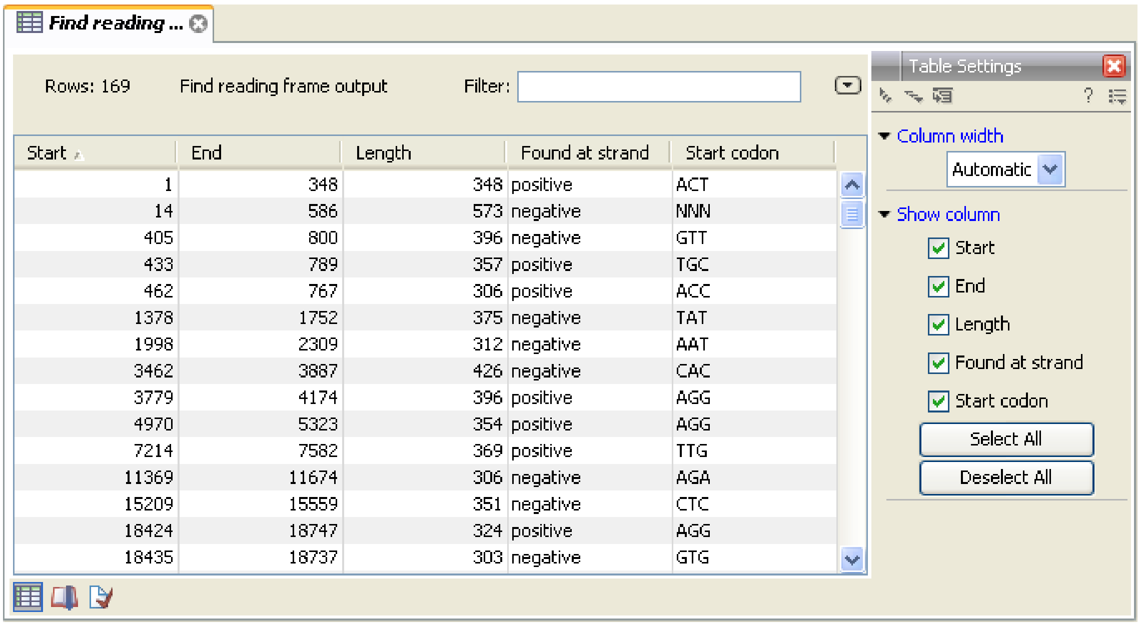
For more information about the graph view, please see section B.

Appendix C

Working with tables

Tables are used in a lot of places in the *CLC Genomics Workbench*. The contents of the tables are of course different depending on the context, but there are some general features for all tables that will be explained in the following.

Figure C.1 shows an example of a typical table. This is the table result of **Find Open Reading Frames** ( ). We will use this table as an example in the following to illustrate the concepts that are relevant for all kinds of tables.



Rows: 169 Find reading frame output Filter:

Start	End	Length	Found at strand	Start codon
1	348	348	positive	ACT
14	586	573	negative	NNN
405	800	396	negative	GTT
433	789	357	positive	TGC
462	767	306	positive	ACC
1378	1752	375	negative	TAT
1998	2309	312	negative	AAT
3462	3887	426	negative	CAC
3779	4174	396	positive	AGG
4970	5323	354	positive	AGG
7214	7582	369	positive	TTG
11369	11674	306	negative	AGA
15209	15559	351	negative	CTC
18424	18747	324	positive	AGG
18435	18737	303	negative	GTG

☑ Start
☑ End
☑ Length
☑ Found at strand
☑ Start codon

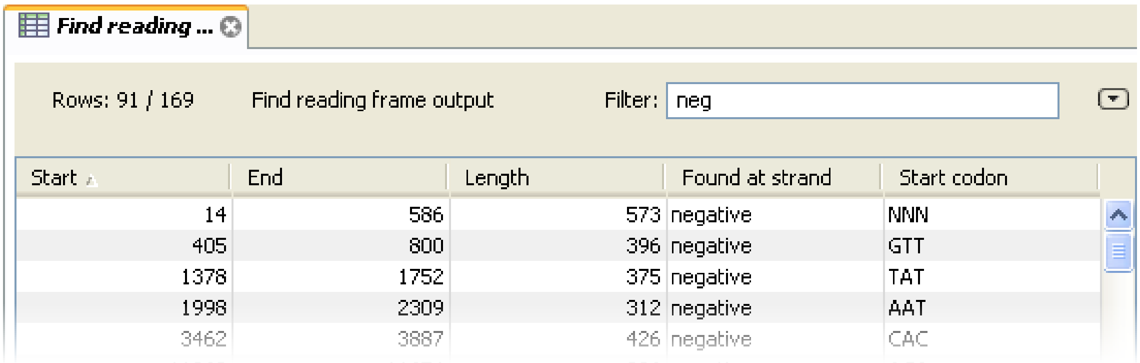
Figure C.1: *A table showing open reading frames.*

First of all, the columns of the table are listed in the **Side Panel** to the right of the table. By clicking the checkboxes you can hide/show the columns in the table.

Furthermore, you can **sort** the table by clicking on the column headers. (Pressing Ctrl - ⌘ on Mac - while you click will refine the existing sorting).

C.1 Filtering tables

The final concept to introduce is **Filtering**. The table filter as an advanced and a simple mode. The simple mode is the default and is applied simply by typing text or numbers (see an example in figure C.2).



Rows: 91 / 169   Find reading frame output   Filter: neg

Start	End	Length	Found at strand	Start codon
14	586	573	negative	NNN
405	800	396	negative	GTT
1378	1752	375	negative	TAT
1998	2309	312	negative	AAT
3462	3887	426	negative	CAC

Figure C.2: *Typing "neg" in the filter in simple mode.*

Typing "neg" in the filter will only show the rows where "neg" is part of the text in any of the columns (also the ones that are not shown). The text does not have to be in the beginning, thus "ega" would give the same result. This simple filter works fine for fast, textual and non-complicated filtering and searching.

However, if you wish to make use of numerical information or make more complex filters, you can switch to the advanced mode by clicking the **Advanced filter** (▾) button. The advanced filter is structure in a different way: First of all, you can have more than one criterion in the filter. Criteria can be added or removed by clicking the **Add** (➕) or **Remove** (✖) buttons. At the top, you can choose whether all the criteria should be fulfilled (**Match all**), or if just one of the needs to be fulfilled (**Match any**).

For each filter criterion, you first have to select which column it should apply to. Next, you choose an operator. For numbers, you can choose between:

- **=** (equal to)
- **<** (smaller than)
- **>** (greater than)
- **<>** (not equal to)
- **abs. value <** (absolute value smaller than. This is useful if it doesn't matter whether the number is negative or positive)
- **abs. value >** (absolute value greater than. This is useful if it doesn't matter whether the number is negative or positive)

For text-based columns, you can choose between:

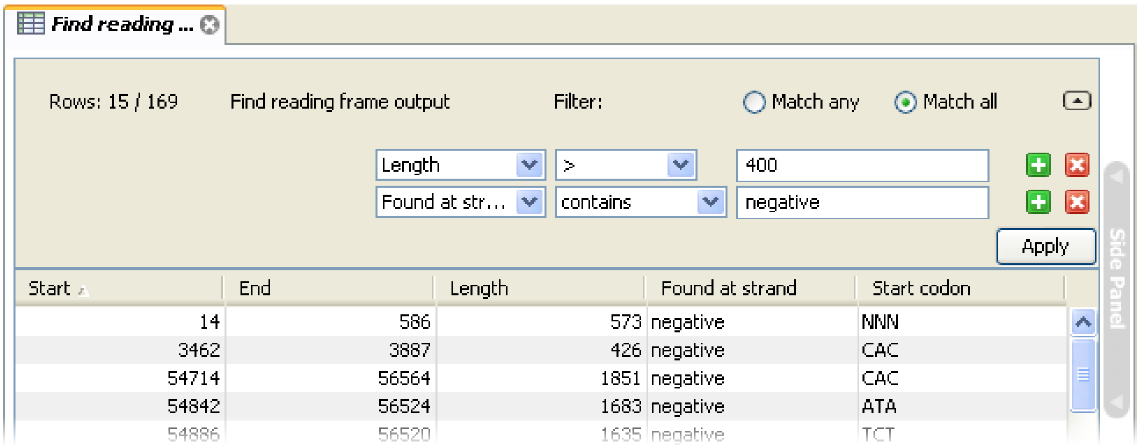
- **contains** (the text does not have to be in the beginning)
- **doesn't contain**

- **=** (the whole text in the table cell has to match, also lower/upper case)

Once you have chosen an operator, you can enter the text or numerical value to use.

If you wish to reset the filter, simply remove (☒) all the search criteria. Note that the last one will not disappear - it will be reset and allow you to start over.

Figure C.3 shows an example of an advanced filter which displays the open reading frames larger than 400 that are placed on the negative strand.



Rows: 15 / 169 Find reading frame output Filter: ○ Match any ● Match all

Length > 400

Found at str... contains negative

Start	End	Length	Found at strand	Start codon
14	586	573	negative	NNN
3462	3887	426	negative	CAC
54714	56564	1851	negative	CAC
54842	56524	1683	negative	ATA
54886	56520	1635	negative	TCT

Figure C.3: *The advanced filter showing open reading frames larger than 400 that are placed on the negative strand.*

Both for the simple and the advanced filter, there is a counter at the upper left corner which tells you the number of rows that pass the filter (91 in figure C.2 and 15 in figure C.3).

Appendix D

BLAST databases

Several databases are available at NCBI, which can be selected to narrow down the possible BLAST hits.

D.1 Peptide sequence databases

- **nr.** Non-redundant GenBank CDS translations + PDB + SwissProt + PIR + PRF, excluding those in env_nr.
- **refseq.** Protein sequences from NCBI Reference Sequence project http://www.ncbi.nlm.nih.gov/RefSeq/.
- **swissprot.** Last major release of the SWISS-PROT protein sequence database (no incremental updates).
- **pat.** Proteins from the Patent division of GenBank.
- **pdb.** Sequences derived from the 3-dimensional structure records from the Protein Data Bank http://www.rcsb.org/pdb/.
- **env_nr.** Non-redundant CDS translations from env_nt entries.
- **month.** All new or revised GenBank CDS translations + PDB + SwissProt + PIR + PRF released in the last 30 days..

D.2 Nucleotide sequence databases

- **nr.** All GenBank + EMBL + DDBJ + PDB sequences (but no EST, STS, GSS, or phase 0, 1 or 2 HTGS sequences). No longer "non-redundant" due to computational cost.
- **refseq_rna.** mRNA sequences from NCBI Reference Sequence Project.
- **refseq_genomic.** Genomic sequences from NCBI Reference Sequence Project.
- **est.** Database of GenBank + EMBL + DDBJ sequences from EST division.
- **est_human.** Human subset of est.

- **est_mouse.** Mouse subset of est.
- **est_others.** Subset of est other than human or mouse.
- **gss.** Genome Survey Sequence, includes single-pass genomic data, exon-trapped sequences, and Alu PCR sequences.
- **htgs.** Unfinished High Throughput Genomic Sequences: phases 0, 1 and 2. Finished, phase 3 HTG sequences are in nr.
- **pat.** Nucleotides from the Patent division of GenBank.
- **pdb.** Sequences derived from the 3-dimensional structure records from Protein Data Bank. They are NOT the coding sequences for the corresponding proteins found in the same PDB record.
- **month.** All new or revised GenBank+EMBL+DDBJ+PDB sequences released in the last 30 days.
- **alu.** Select Alu repeats from REPBASE, suitable for masking Alu repeats from query sequences. See "Alu alert" by Claverie and Makalowski, Nature 371: 752 (1994).
- **dbsts.** Database of Sequence Tag Site entries from the STS division of GenBank + EMBL + DDBJ.
- **chromosome.** Complete genomes and complete chromosomes from the NCBI Reference Sequence project. It overlaps with refseq_genomic.
- **wgs.** Assemblies of Whole Genome Shotgun sequences.
- **env_nt.** Sequences from environmental samples, such as uncultured bacterial samples isolated from soil or marine samples. The largest single source is Sagarsso Sea project. This does overlap with nucleotide nr.

D.3 SNP BLAST databases

The list of databases for **SNP Annotation Using BLAST** () is available at http://www.ncbi.nlm.nih.gov/staff/tao/URLAPI/remote_blastdblist.html#8.

D.4 Adding more databases

Besides the databases that are part of the default configuration, you can add more databases located at NCBI by configuring files in the Workbench installation directory.

The list of databases that can be added is here: http://www.ncbi.nlm.nih.gov/staff/tao/URLAPI/remote_blastdblist.html.

In order to add a new database, find the `settings` folder in the Workbench installation directory (e.g. `C:\Program files\CLC Genomics Workbench 4`). Download unzip and place the following files in this directory to replace the built-in list of databases:

- Nucleotide databases: http://www.clcbio.com/wbsettings/NCBI_BlastNucleotideDataba properties.zip

- Protein databases: http://www.clcbio.com/wbsettings/NCBI_BlastProteinDatabases.properties.zip

Open the file you have downloaded into thesettings folder, e.g. NCBI_BlastProteinDatabases.proper in a text editor and you will see the contents look like this:

```
nr[clcdefault] = Non-redundant protein sequences
refseq_protein = Reference proteins
swissprot = Swiss-Prot protein sequences
pat = Patented protein sequences
pdb = Protein Data Bank proteins
env_nr = Environmental samples
month = New or revised GenBank sequences
```

Simply add another database as a new line with the first item being the database name taken from http://www.ncbi.nlm.nih.gov/staff/tao/URLAPI/remote_blastdblist.html and the second part is the name to display in the Workbench. Restart the Workbench, and the new database will be visible in the BLAST dialog.

Appendix E

Proteolytic cleavage enzymes

Most proteolytic enzymes cleave at distinct patterns. Below is a compiled list of proteolytic enzymes used in *CLC Genomics Workbench*.

Name	P4	P3	P2	P1	P1’	P2’
Cyanogen bromide (CNBr)	-	-	-	M	-	-
Asp-N endopeptidase	-	-	-	-	D	-
Arg-C	-	-	-	R	-	-
Lys-C	-	-	-	K	-	-
Trypsin	-	-	-	K, R	not P	-
Trypsin	-	-	W	K	P	-
Trypsin	-	-	M	R	P	-
Trypsin*	-	-	C, D	K	D	-
Trypsin*	-	-	C	K	H, Y	-
Trypsin*	-	-	C	R	K	-
Trypsin*	-	-	R	R	H,R	-
Chymotrypsin-high spec.	-	-	-	F, Y	not P	-
Chymotrypsin-high spec.	-	-	-	W	not M, P	-
Chymotrypsin-low spec.	-	-	-	F, L, Y	not P	-
Chymotrypsin-low spec.	-	-	-	W	not M, P	-
Chymotrypsin-low spec.	-	-	-	M	not P, Y	-
Chymotrypsin-low spec.	-	-	-	H	not D, M, P, W	-
o-Iodosobenzoate	-	-	-	W	-	-
Thermolysin	-	-	-	not D, E	A, F, I, L, M or V	-
Post-Pro	-	-	H, K, R	P	not P	-
Glu-C	-	-	-	E	-	-
Asp-N	-	-	-	-	D	-
Proteinase K	-	-	-	A, E, F, I, L, T, V, W, Y	-	-
Factor Xa	A, F, G, I, L, T, V, M	D,E	G	R	-	-
Granzyme B	I	E	P	D	-	-
Thrombin	-	-	G	R	G	-
Thrombin	A, F, G, I, L, T, V, M	A, F, G, I, L, T, V, W, A	P	R	not D, E	not D, E
TEV (Tobacco Etch Virus)	-	Y	-	Q	G, S	-

Appendix F

Restriction enzymes database configuration

CLC Genomics Workbench uses enzymes from the **REBASE** restriction enzyme database at http://rebase.neb.com. If you wish to add enzymes to this list, you can do this by manually using the procedure described here.

Note! Please be aware that this process needs to be handled carefully, otherwise you may have to re-install the Workbench to get it to work.

First, download the following file and unzip it: http://www.clcbio.com/wbsettings/link_emboss_e_custom.zip. In the Workbench installation folder under settings, create a folder named rebase and place the extracted link_emboss_e_custom file here. Open the file in a text editor. The top of the file contains information about the format, and at the bottom there are two example enzymes that you should replace with your own.

Restart the Workbench to have the changes take effect.

Appendix G

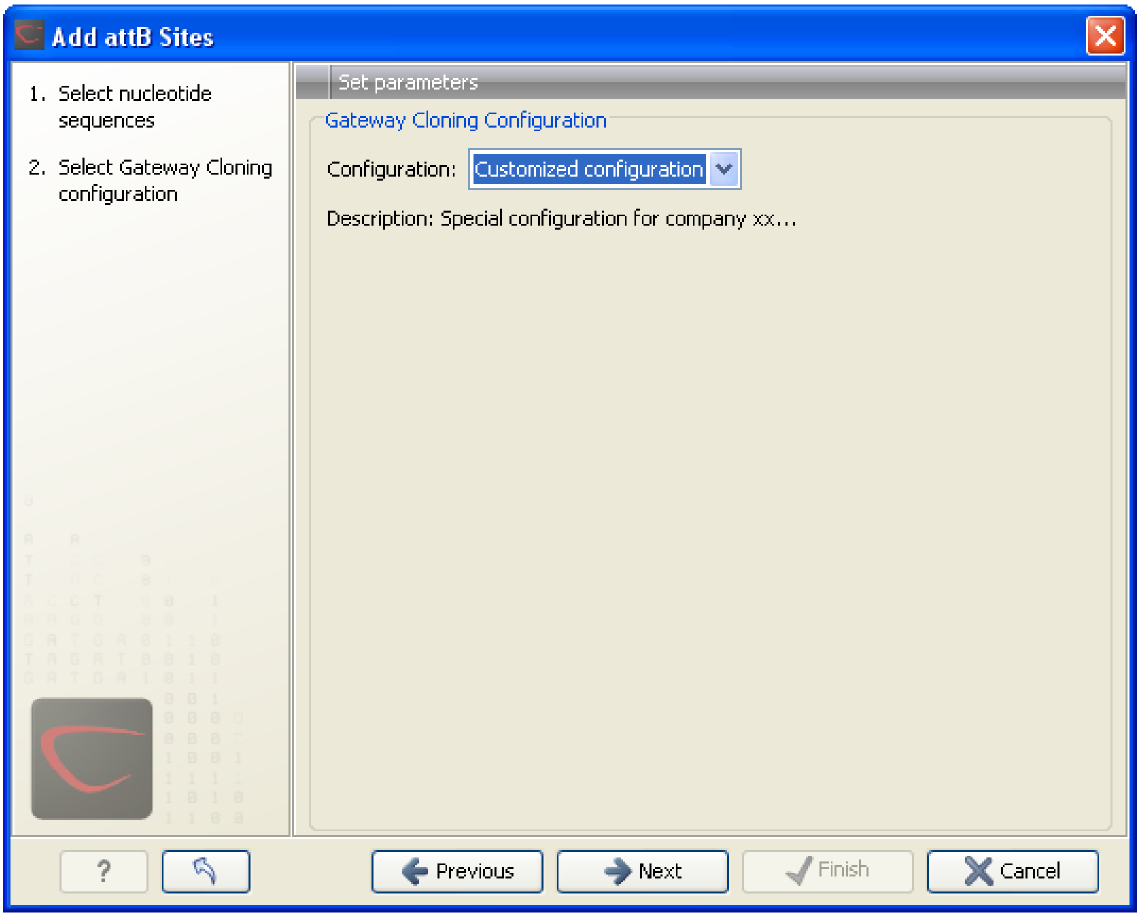
Technical information about modifying Gateway cloning sites

The *CLC Genomics Workbench* comes with a pre-defined list of Gateway recombination sites. These sites and the recombination logics can be modified by downloading and editing a properties file. Note that this is a technical procedure only needed if the built-in functionality is not sufficient for your needs.

The properties file can be downloaded from `http://www.clcbio.com/wbsettings/gatewaycloning.zip`. Extract the file included in the zip archive and save it in the `settings` folder of the Workbench installation folder. The file you download contains the standard configuration. You should thus update the file to match your specific needs. See the comments in the file for more information.

The name of the properties file you download is `gatewaycloning.1.properties`. You can add several files with different configurations by giving them a different number, e.g. `gatewaycloning.2.properties` and so forth. When using the Gateway tools in the Workbench, you will be asked which configuration you want to use (see figure G.1).

Note that you need to restart the Workbench after editing or adding the files for the changes to take effect.


Add attB Sites
1. Select nucleotide sequences
2. Select Gateway Cloning configuration
Set parameters
Gateway Cloning Configuration
?
Previous
Next
Finish
Cancel

Configuration: Customized configuration

Description: Special configuration for company xx...

Figure G.1: *Selecting between different gateway cloning configurations.*

Appendix H

IUPAC codes for amino acids

(Single-letter codes based on International Union of Pure and Applied Chemistry)

The information is gathered from: http://www.ebi.ac.uk/2can/tutorials/aa.html

One-letter abbreviation	Three-letter abbreviation	Description
A	Ala	Alanine
R	Arg	Arginine
N	Asn	Asparagine
D	Asp	Aspartic acid
C	Cys	Cysteine
Q	Gln	Glutamine
E	Glu	Glutamic acid
G	Gly	Glycine
H	His	Histidine
J	Xle	Leucine or Isoleucineucine
L	Leu	Leucine
I	ILe	Isoleucine
K	Lys	Lysine
M	Met	Methionine
F	Phe	Phenylalanine
P	Pro	Proline
O	Pyl	Pyrrolysine
U	Sec	Selenocysteine
S	Ser	Serine
T	Thr	Threonine
W	Trp	Tryptophan
Y	Tyr	Tyrosine
V	Val	Valine
B	Asx	Aspartic acid or Asparagine Asparagine
Z	Glx	Glutamic acid or Glutamine Glutamine
X	Xaa	Any amino acid

Appendix I

IUPAC codes for nucleotides

(Single-letter codes based on International Union of Pure and Applied Chemistry)

The information is gathered from: http://www.iupac.org and http://www.ebi.ac.uk/2can/tutorials/aa.html.

Code	Description
A	Adenine
C	Cytosine
G	Guanine
T	Thymine
U	Uracil
R	Purine (A or G)
Y	Pyrimidine (C, T, or U)
M	C or A
K	T, U, or G
W	T, U, or A
S	C or G
B	C, T, U, or G (not A)
D	A, T, U, or G (not C)
H	A, T, U, or C (not G)
V	A, C, or G (not T, not U)
N	Any base (A, C, G, T, or U)

Appendix J

Formats for import and export

J.1 List of bioinformatic data formats

Below is a list of bioinformatic data formats, i.e. formats for importing and exporting sequences, alignments and trees.

J.1.1 Sequence data formats

Note that high-throughput sequencing data formats from Illumina Genome Analyzer, SOLiD, 454 and also high-throughput fasta and trace files are imported using a special import as described in section 19.1. These data can also be exported in fastq format (using NCBI/Sanger Phred quality scores).

File type	Suffix	Import	Export	Description
FASTA	.fsa/.fasta	X	X	Simple format, name & description
AB1	.ab1	X		Including chromatograms
ABI	.abi	X		Including chromatograms
CLC	.clc	X	X	Rich format including all information
Clone Manager	.cm5	X		
CSV export	.csv		X	Annotations in csv format
CSV import	.csv	X		One sequence per line: name; description(optional); sequence
DNAstrider	.str/.strider	X	X	
DS Gene	.bsml	X		
Embl	.embl	X	X	Only nucleotide sequence
GCG sequence	.gcg	X		Rich information incl. annotations
GenBank	.gbk/.gb/.gp	X	X	Rich information incl. annotations
Gene Construction Kit	.gck	X		
Lasergene	.pro/.seq	X		
Nexus	.nxs/.nexus	X	X	
Phred	.phd	X		Including chromatograms
PIR (NBRF)	.pir	X		Simple format, name & description
Raw sequence	any	X		Only sequence (no name)
SCF2	.scf	X		Including chromatograms
SCF3	.scf	X		Including chromatograms
Staden	.sdn	X		
Swiss-Prot	.swp	X	X	Rich information (only proteins)
Tab delimited text	.txt		X	Annotations in tab delimited text format
Vector NTI archives	.ma4/.pa4/.oa4	X		Archives in rich format
Vector NTI Database		X		Special import full database
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gzip./.tar	X		Contained files/folder structure

J.1.2 Read mapping formats

File type	Suffix	Import	Export	Description
ACE	.ace	X	X	No chromatogram or quality score
BAM (Compressed version of SAM)	.bam	X	X	See details in section 19.1.7
CLC	.clc	X	X	Rich format including all information
CLC Assembly File	.cas	X		Output from the CLC Assembly Cell
SAM (Sequence Alignment/Map)	.sam	X	X	See details in section 19.1.7
Tabular assembly (e.g. ELAND)	.aln/.txt/.bed	X		See details in section 19.1.8
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip	X		Contained files/folder structure

Special notes for SAM/BAM export:

- Broken pairs on RNA-seq samples are always exported as single reads. Broken pairs are only exported as paired if both reads map on the same gene.

J.1.3 Alignment formats

File type	Suffix	Import	Export	Description
CLC	.clc	X	X	Rich format including all information
Clustal Alignment	.aln	X	X	
GCG Alignment	.msf	X	X	
Nexus	.nxs/.nexus	X	X	
Phylip Alignment	.phy	X	X	
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gzip./.tar	X		Contained files/folder structure

J.1.4 Tree formats

File type	Suffix	Import	Export	Description
CLC	.clc	X	X	Rich format including all information
Newick	.nwk	X	X	
Nexus	.nxs/.nexus	X	X	
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gzip./.tar	X		Contained files/folder structure

J.1.5 Expression data formats

Read about technical details of these data formats in section K.

File type	Suffix	Import	Export	Description
Affymetrix CHP	.chp/.psi	X		Expression values and annotations
Affymetrix pivot/metric	.txt/.csv	X		Gene-level expression values
Affymetrix NetAffx	.csv	X		Annotations
CLC	.clc	X	X	Rich format including all information
CSV	.csv		X	Samples and experiments,
Excel	.xls/.xlsx		X	All tables and reports
Generic	.txt/.csv	X		expression values
Generic	.txt/.csv	X		annotations
GEO soft sample/series	.txt/.csv	X		Expression values
Illumina	.txt	X		Expression values and annotations
Tab delimited	.txt		X	Samples and experiments,
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gzip./.tar	X		Contained files/folder structure

J.1.6 Miscellaneous formats

File type	Suffix	Import	Export	Description
BLAST Database	.phr/.nhr	X		Link to database imported
CLC	.clc	X	X	Rich format including all information
CSV	.csv		X	All tables
Excel	.xls/.xlsx		X	All tables and reports
GFF	.gff	X	X	See `http://www.clcbio.com/annotate-with-gff`
mmCIF	.cif	X		3D structure
PDB	.pdb	X		3D structure
RNA structures	.ct, .col, .rnaml/.xml	x		Secondary structure for RNA
Tab delimited	.txt		X	All tables
Text	.txt	X	X	All data in a textual format
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gzip./.tar	X		Contained files/folder structure

Note! The Workbench can import 'external' files, too. This means that all kinds of files can be imported and displayed in the **Navigation Area**, but the above mentioned formats are the only ones whose *contents* can be shown in the Workbench.

J.2 List of graphics data formats

Below is a list of formats for exporting graphics. All data displayed in a graphical format can be exported using these formats. Data represented in lists and tables can only be exported in .pdf format (see section 7.3 for further details).

Format	**Suffix**	**Type**
Portable Network Graphics	.png	bitmap
JPEG	.jpg	bitmap
Tagged Image File	.tif	bitmap
PostScript	.ps	vector graphics
Encapsulated PostScript	.eps	vector graphics
Portable Document Format	.pdf	vector graphics
Scalable Vector Graphics	.svg	vector graphics

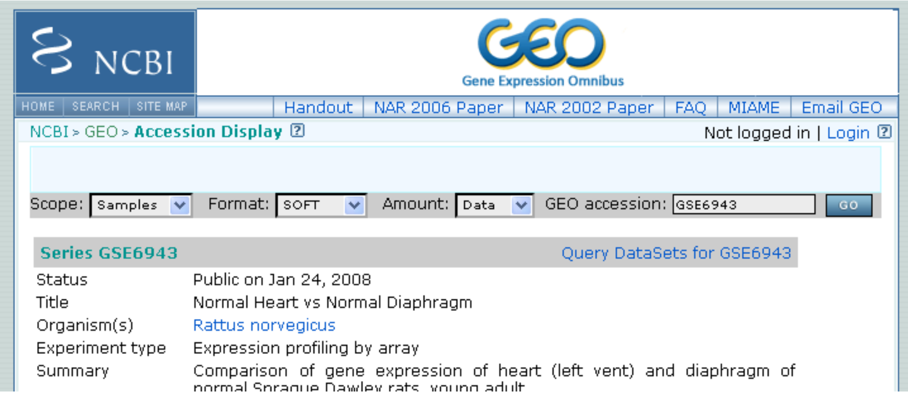
Appendix K

Expression data formats

Below you find descriptions of the microarray data formats that are supported by *CLC Genomics Workbench*. Note that we for some platforms support both expression data and annotation data.

K.1 GEO (Gene Expression Omnibus)

The GEO (Gene Expression Omnibus) sample and series formats are supported. Figure K.1 shows how to download the data from GEO in the right format. GEO is located at `http://www.ncbi.nlm.nih.gov/geo/`.


NCBI
HOME | SEARCH | SITE MAP
GEO
Gene Expression Omnibus
Handout | NAR 2006 Paper | NAR 2002 Paper | FAQ | MIAME | Email GEO
NCBI > GEO > Accession Display
Not logged in | Login
GO

Scope: Samples Format: SOFT Amount: Data GEO accession: GSE6943

Series GSE6943 Query DataSets for GSE6943

Status	Public on Jan 24, 2008
Title	Normal Heart vs Normal Diaphragm
Organism(s)	Rattus norvegicus
Experiment type	Expression profiling by array
Summary	Comparison of gene expression of heart (left vent) and diaphragm of normal Sprague Dawley rats, young adult

Figure K.1: *Selecting Samples, SOFT and Data before clicking go will give you the format supported by the* **CLC Genomics Workbench***.*

The GEO sample files are tab-delimited .txt files. They have three required lines:

```
^SAMPLE = GSM21610
!sample_table_begin
...
!sample_table_end
```

The first line should start with `^SAMPLE =` followed by the sample name, the line `!sample_table_begin` and the line `!sample_table_end`. Between the `!sample_table_begin` and `!sample_table_end`, lines are the column contents of the sample.

Note that GEO sample importer will also work for concatenated GEO sample files — allowing multiple samples to be imported in one go. Download a sample file containing concatenated sample files here:
http://www.clcbio.com/madata/GEOSampleFilesConcatenated.txt

Below you can find examples of the formatting of the GEO formats.

K.1.1 GEO sample file, simple

This format is very simple and includes two columns: one for feature id (e.g. gene name) and one for the expression value.

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF    VALUE
id1       105.8
id2       32
id3       50.4
id4       57.8
id5       2914.1
!sample_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSampleFileSimple.txt

K.1.2 GEO sample file, including present/absent calls

This format includes an extra column for absent/present calls that can also be imported.

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF    VALUE    ABS_CALL
id1       105.8    M
id2       32       A
id3       50.4     A
id4       57.8     A
id5       2914.1   P
!sample_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSampleFileAbsentPresent.txt

K.1.3 GEO sample file, including present/absent calls and p-values

This format includes two extra columns: one for absent/present calls and one for absent/present call p-values, that can also be imported.

```
^SAMPLE = GSM21610
```

```
!sample_table_begin
ID_REF     VALUE      ABS_CALL      DETECTION P-VALUE
id1        105.8      M             0.00227496
id2        32         A             0.354441
id3        50.4       A             0.904352
id4        57.8       A             0.937071
id5        2914.1     P             6.02111e-05
!sample_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSampleFileAbsentPresentCallAndPValue.txt

K.1.4 GEO sample file: using absent/present call and p-value columns for sequence information

The workbench assumes that if there is a third column in the GEO sample file then it contains present/absent calls and that if there is a fourth column then it contains p-values for these calls. This means that the contents of the third column is assumed to be text and that of the fourth column a number. As long as these two basic requirements are met, the sample should be recognized and interpreted correctly.

You can thus use these two columns to carry additional information on your probes. The absent/present column can be used to carry additional information like e.g. sequence tags as shown below:

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF       VALUE      ABS_CALL
id1         105.8       AAA
id2         32          AAC
id3         50.4        ATA
id4         57.8        ATT
id5         2914.1      TTA
!sample_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSampleFileSimpleSequenceTag.txt

Or, if you have multiple probes per sequence you could use the present/absent column to hold the sequence name and the p-value column to hold the interrogation position of your probes:

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF     VALUE     ABS_CALL     DETECTION P-VALUE
probe1     755.07    seq1         1452
probe2     587.88    seq1         497
probe3     716.29    seq1         1447
probe4     1287.18   seq2         1899
!sample_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSampleFileSimpleSequenceTagAndProbe.txt

K.1.5 GEO series file, simple

The series file includes expression values for multiple samples. Each of the samples in the file will be represented by its own element with the sample name. The first row lists the sample names.

```
!Series_title "Myb specificity determinants"
!series_matrix_table_begin
"ID_REF" "GSM21610" "GSM21611" "GSM21612"
"id1"      2541      1781.8     1804.8
"id2"      11.3      621.5     50.2
"id3"      61.2      149.1     22
"id4"      55.3      328.8     97.2
"id5"        183.8       378.3      423.2
!series_matrix_table_end
```

Download the sample file here:
http://www.clcbio.com/madata/GEOSeriesFile.txt

K.2 Affymetrix GeneChip

For Affymetrix, three types of files are currently supported: Affymetrix .CHP files, Affymetrix NetAffx annotation files and tab-delimited pivot or metrics files. Affymetrix .CEL files are currently not supported. However, the Bioconductor R package 'affy' allows you to preprocess the .CEL files and export a txt file containing a table of estimated gene-level expression values in three lines of code:

```
library(affy) # loading Bioconductor library 'affy'
data=ReadAffy() # probe-level data import
eset=rma(data) # probe-level data pre-processing using 'rma'
write.exprs(eset,file="evals.txt") # writing gene expression levels to 'evals-txt'
```

The exported txt file (evals.txt) can be imported into the workbench using the Generic expression data table format importer (see section K.5; you can just 'drag-and-drop' it in). In R, you should have all the CEL files you wish to process in your working directory and the file 'evals.txt' will be written to that directory.

K.2.1 Affymetrix CHP expression files

The Affymetrix scanner software produces a number of files when a GeneChip is scanned. Two of these are the .CHP and the .CEL files. These are binary files with native Affymetrix formats. The Affymetrix GeneChips contain a number of probes for each gene (typically between 22 and 40). The .CEL file contains the probe-level intensities, and the .CHP file contains the gene-level information. The gene-level information has been obtained by the scanner software through postprocessing and summarization of the probe-level intensities.

In order to interpret the probe-level information in the .CEL file, the .CDF file for the type of GeneChip that was used is required. Similarly for the .CHP file: in order to interpret the gene-level information in the .CHP file, the .PSI file for the type of GeneChip that was used is required.

In order to import a .CHP file it is required that the corresponding .PSI file is present in the same folder as the .CHP file you want to import, and furthermore, this must be the only .PSI file that is present there. There are no requirements for the name of the .PSI file. Note that the .PSI file itself will not be imported - it is only used to guide the import of the .CHP file which contains the expression values.

Download example .CHP and .PSI files here (note that these are binary files):
http://www.clcbio.com/madata/AffymetrixCHPandPSI.zip

K.2.2 Affymetrix metrix files

The Affymetrix metrics or pivot files are tab-delimited files that may be exported from the Affymetrix scanner software. The metrics files have a lot of technical information that is only partly used in the Workbench. The feature ids (Probe Set Name), expression values (Used Signal), absent/present call (Detection) and absent/present p-value (Detection p-value) are imported into the Workbench.

Download a small example sample file here:
http://www.clcbio.com/madata/AffymetrixMetrics.txt

K.2.3 Affymetrix NetAffx annotation files

The NetAffx annotation files for Whole-Transcript Expression Gene arrays and 3' IVT Expression Analysis Arrays can be imported and used to annotate experiments as shown in section 20.1.4.

Download a small example annotation file here which includes header information:
http://www.clcbio.com/madata/AffymetrixNetAffxAnnotationFile.csv

K.3 Illumina BeadChip

Both BeadChip expression data files from Illumina's BeadStudio software and the corresponding BeadChip annotation files are supported by *CLC Genomics Workbench*. The formats of the BeadStudio and annotation files have changed somewhat over time and various formats are supported.

K.3.1 Illumina expression data, compact format

An example of this format is shown below:

```
TargetID          AVG_Signal        BEAD_STDEV       Detection
GI_10047089-S     112.5             4.2              0.16903226
GI_10047091-S     127.6             4.8              0.76774194
```

All this information is imported into the Workbench. The AVG_Signal is used as the expression measure.

Download a small sample file here:
http://www.clcbio.com/madata/IlluminaBeadChipCompact.txt

K.3.2 Illumina expression data, extended format

An example of this format is shown below:

```
TargetID        MIN_Signal  AVG_Signal  MAX_Signal  NARRAYS  ARRAY_STDEV  BEAD_STDEV  Avg_NBEADS  Detection
GI_10047089-S   73.7        73.7        73.7        1        NaN          3.4         53          0.05669084
GI_10047091-S   312.7       312.7       312.7       1        NaN          11.1        50          0.99604483
```

All this information is imported into the Workbench. The `AVG_Signal` is used as the expression measure.

Download a small sample file here:
http://www.clcbio.com/madata/IlluminaBeadChipExtended.txt

K.3.3 Illumina expression data, with annotations

An example of this format is shown below:

```
TargetID Accession Symbol Definition Synonym Signal-BG02 DCp32  Detection-BG02 DCp32
GI_10047089-S NM_014332.1 SMPX "Homo sapiens small muscle protein, X-linked (SMPX), mRNA."  -17.6  0.03559657
GI_10047091-S NM_013259.1 NP25 "Homo sapiens neuronal protein (NP25), mRNA." NP22  32.6  0.99604483
GI_10047093-S NM_016299.1 HSP70-4 "Homo sapiens likely ortholog of mouse heat shock protein, 70 kDa 4 (HSP70-4), mRNA."  228.1 1
```

Only the TargetID, Signal and Detection columns will be imported, the remaining columns will be ignored. This means that the annotations are not imported. The `Signal` is used as the expression measure.

Download a small example sample file here:
http://www.clcbio.com/madata/IlluminaBeadStudioWithAnnotations.txt

K.3.4 Illumina expression data, multiple samples in one file

This file format has too much information to show it inline in the text. You can download a small example sample file here:
http://www.clcbio.com/madata/IlluminaBeadStudioMultipleSamples.txt

This file contains data for 18 samples. Each sample has an expression value (the value in the `AVG_Signal` column), a detection p-value, a bead standard deviation and an average bead number column. The workbench recognizes the 18 samples and their columns.

K.3.5 Illumina annotation files

The Workbench supports import of two types of Illumina BeadChip annotation files. These are either comma-separated or tab-delimited .txt files. They can be used to annotate experiments as shown in section 20.1.4.

This file format has too much information to show it inline in the text.

Download a small example annotation file of the first type here:
http://www.clcbio.com/madata/IlluminaBeadChipAnnotation.txt
and the second type here:
http://www.clcbio.com/madata/IlluminaBeadChipAnnotationV2.txt

K.4 Gene ontology annotation files

The Gene ontology web site provides annotation files for a variety of species which can all be downloaded and imported into the *CLC Genomics Workbench*. This can be used to annotate experiments as shown in section 20.1.4. See the complete list including download links at http://www.geneontology.org/GO.current.annotations.shtml.

This is an easy way to annotate your experiment with GO categories.

K.5 Generic expression and annotation data file formats

If you have your expression or annotation data in e.g. Excel and can export the data as a txt file, or if you are able to do some scripting or other manipulations to format your data files, you will be able to import them into the *CLC Genomics Workbench* as a 'generic' expression or annotation data file. There are a few simple requirements that need to be fulfilled to do this as described below.

K.5.1 Generic expression data table format

The *CLC Genomics Workbench* will import a tab, semicolon or comma-separated .txt or .csv file as expression array samples if the following requirements are met:

1. the first non-empty line of the file contains text. All entries, except the first, will be used as sample names
2. the following (non-empty) lines contain the same number of entries as the first non-empty line. The requirements to these are that the first entry should be a string (this will be used as the feature ID) and the remaining entries should contain numbers (which will be used as expression values — one per sample). Empty entries are not allowed, but NaN values are allowed.

An example of this format is shown below:

```
FeatureID; sample1; sample2; sample3
gene1; 200;300;23
gene2; 210;30;238
gene3; 230;50;23
gene4; 50;100;235
gene5; 200;300;23
gene6; 210;30;238
gene7; 230;50;23
gene8; 50;100;235
```

This will be imported as three samples with eight genes in each sample.

Download a this example as a file here:
http://www.clcbio.com/madata/CustomExpressionData.txt

K.5.2 Generic annotation file for expression data format

The *CLC Genomics Workbench* will import a tab, semicolon or comma-separated .txt or .csv file as an annotation file if the following is met:

1. It has a line which can serve as a valid header line. In order to do this, the line should have a number of headers where at least two are among the valid column headers in the **Column header** column below.
2. It contains one of the `PROBE_ID` headers (that is: 'Probe Set ID', 'Feature ID', 'ProbeID' or 'Probe_Id').

The importer will import an annotation table with a column for each of the valid column headers (those in the **Column header** column below). Columns with invalid headers will be ignored.

Note that some column headers are alternatives so that only one of the alternative columns headers should be used.

Also note that when adding annotations from an annotation file onto an experiment the file contents are merged onto the experiment by matching the entries in a user specified annotation column in the annotation file to the entries in the feature id column of the experiment. It will thus typically be useful to include a column in your annotation file in which the entries are a subset of, or identical to, the entries in the Feature Id column of the experiment you want to annotate.

A simple example of an annotation file is shown here:

```
"Probe Set ID","Gene Symbol","Gene Ontology Biological Process"
"1367452_at","Sumo2","0006464 // protein modification process //  not recorded"
"1367453_at","Cdc37","0051726 // regulation of cell cycle //  not recorded"
"1367454_at","Copb2","0006810 // transport //  ///  0016044 // membrane organization // "
```

Download this example plus a more elaborate one here:
http://www.clcbio.com/madata/SimpleCustomAnnotation.csv
http://www.clcbio.com/madata/FullCustomAnnotation.csv

To meet requirements imposed by special functionalities in the workbench, there are a number of further restrictions on the contents in the entries of the columns:

Download sequence functionality In the experiment table, you can click a button to download sequence. This uses the contents of the `PUBLIC_ID` column, so this column must be present for the action to work and should contain the NCBI accession number.

Annotation tests The annotation tests can make use of several entries in a column as long as a certain format is used. The tests assume that entries are separated by /// and it interprets all that appears before // as the actual entry and all that appears after // within an entry as comments. Example:

```
/// 0000001 //  comment1  /// 0000008 // comment2 /// 0003746 //  comment3
```

The annotation tests will interpret this as three entries (0000001, 0000008, and 0003746) with the according comments.

The most common column headers are summarized below:

Column header in imported file (alternatives separated by commas)	Label in experiment table	Description (tool tip)
Probe Set ID, Feature ID, ProbeID, Probe_Id, transcript_cluster_id	Feature ID	Probe identifier tag
Representative Public ID, Public identifier tag, GenbankAccession	Public identifier tag	Representative public ID
Gene Symbol, GeneSymbol	Gene symbol	Gene symbol
Gene Ontology Biological Process, Ontology_Process, GO_biological_process	GO biological process	Gene Ontology biological process
Gene Ontology Cellular Component, Ontology_Component, GO_cellular_component	GO cellular component	Gene Ontology cellular componen
Gene Ontology Molecular Function, Ontology_Function, GO_molecular_function	GO molecular function	Gene Ontology molecular functio
Pathway	Pathway	Pathway

The full list of possible column headers:

Column header in imported file (alternatives separated by commas)	**Label in experiment table**	**Description (tool tip)**
Species Scientific Name, Species Name, Species	Species name	Scientific species name
GeneChip Array	Gene chip array	Gene Chip Array name
Annotation Date	Annotation date	Date of annotation
Sequence Type	Sequence type	Type of sequence
Sequence Source	Sequence source	Source from which sequence was obtained
Transcript ID(Array Design), Transcript	Transcript ID	Transcript identifier tag
Target Description	Target description	Target description
Archival UniGene Cluster	Archival UniGene cluster	Archival UniGene cluster
UniGene ID, UniGeneID, Unigene_ID, unigene	UniGene ID	UniGene identifier tag
Genome Version	Genome version	Version of genome on which annotation is based
Alignments	Alignments	Alignments
Gene Title	Gene title	Gene title
geng_assignments	Gene assignments	Gene assignments
Chromosomal Location	Chromosomal location	Chromosomal location
Unigene Cluster Type	UniGene cluster type	UniGene cluster type
Ensemble Ensembl	Ensembl	
Entrez Gene, EntrezGeneID, Entrez_Gene_ID	Entrez gene	Entrez gene
SwissProt	SwissProt	SwissProt
EC	EC	EC
OMIM	OMIM	Online Mendelian Inheritance in Man
RefSeq Protein ID	RefSeq protein ID	RefSeq protein identifier tag
RefSeq Transcript ID	RefSeq transcript ID	RefSeq transcript identifier tag
FlyBase	FlyBase	FlyBase
AGI	AGI	AGI
WormBase	WormBase	WormBase
MGI Name	MGI name	MGI name
RGD Name	RGD name	RGD name
SGD accession number	SGD accession number	SGD accession number
InterPro	InterPro	InterPro
Trans Membrane	Trans membrane	Trans membrane
QTL	QTL	QTL
Annotation Description	Annotation description	Annotation description
Annotation Transcript Cluster	Annotation transcript cluster	Annotation transcript cluster
Transcript Assignments	Transcript assignments	Trancript assignments
mrna_assignments	mRNA assignments	mRNA assignments
Annotation Notes	Annotation notes	Annotation notes
GO, Ontology	Go annotations	Go annotations
Cytoband	Cytoband	Cytoband
PrimaryAccession	Primary accession	Primary accession
RefSeqAccession	RefSeq accession	RefSeq accession
GeneName	Gene name	Gene name
TIGRID	TIGR Id	TIGR Id
Description	Description	Description
GenomicCoordinates	Genomic coordinates	Genomic coordinates
Search_key	Search key	Search key
Target	Target	Target
Gid, GI	Genbank identifier	Genbank identifier
Accession	GenBank accession	GenBank accession
Symbol	Gene symbol	Gene symbol
Probe_Type	Probe type	Probe type
crosshyb_type	Crosshyb type	Crosshyb type
category	category	category
Start, Probe_Start	Start	Start
Stop	Stop	Stop
Definition	Definition	Definition
Synonym, Synonyms	Synonym	Synonym
Source	Source	Source
Source_Reference_ID	Source reference id	Source reference id
RefSeq_ID	Reference sequence id	Reference sequence id
ILMN_Gene	Illumina Gene	Illumina Gene
Protein_Product	Protein product	Protein product
protein_domains	Protein domains	Protein domains
Array_Address_Id	Array adress id	Array adress id
Probe_Sequence	Sequence	Sequence
seqname	Seqname	Seqname
Chromosome	Chromosome	Chromosome
strand	Strand	Strand
Probe_Chr_Orientation	Probe chr orientation	Probe chr orientation
Probe_Coordinates	Probe coordinates	Probe coordinates
Obsolete_Probe_Id	Obsolete probe id	Obsolete probe id

Appendix L

Custom codon frequency tables

You can edit the list of codon frequency tables used by *CLC Genomics Workbench*.

Note! Please be aware that this process needs to be handled carefully, otherwise you may have to re-install the Workbench to get it to work.

In the Workbench installation folder under `res`, there is a folder named `codonfreq`. This folder contains all the codon frequency tables organized into subfolders in a hierarchy. In order to change the tables, you simply add, delete or rename folders and the files in the folders. If you wish to add new tables, please use the existing ones as template.

Restart the Workbench to have the changes take effect.

Please note that when updating the Workbench to a new version, this information is not preserved. This means that you should keep this information in a separate place as back-up. (The ability to change the tables is mainly aimed at centrally deployed installations of the Workbench).

Bibliography

[Akmaev and Wang, 2004] Akmaev, V. R. and Wang, C. J. (2004). Correction of sequence-based artifacts in serial analysis of gene expression. *Bioinformatics*, 20(8):1254–1263.

[Allison et al., 2006] Allison, D., Cui, X., Page, G., and Sabripour, M. (2006). Microarray data analysis: from disarray to consolidation and consensus. *NATURE REVIEWS GENETICS*, 7(1):55.

[Altschul and Gish, 1996] Altschul, S. F. and Gish, W. (1996). Local alignment statistics. *Methods Enzymol*, 266:460–480.

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

[Altshuler et al., 2000] Altshuler, D., Pollara, V. J., Cowles, C. R., Etten, W. J. V., Baldwin, J., Linton, L., and Lander, E. S. (2000). An snp map of the human genome generated by reduced representation shotgun sequencing. *Nature*, 407(6803):513–516.

[Andrade et al., 1998] Andrade, M. A., O'Donoghue, S. I., and Rost, B. (1998). Adaptation of protein surfaces to subcellular location. *J Mol Biol*, 276(2):517–525.

[Bachmair et al., 1986] Bachmair, A., Finley, D., and Varshavsky, A. (1986). In vivo half-life of a protein is a function of its amino-terminal residue. *Science*, 234(4773):179–186.

[Baggerly et al., 2003] Baggerly, K., Deng, L., Morris, J., and Aldaz, C. (2003). Differential expression in SAGE: accounting for normal between-library variation. *Bioinformatics*, 19(12):1477–1483.

[Bateman et al., 2004] Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L. L., Studholme, D. J., Yeats, C., and Eddy, S. R. (2004). The Pfam protein families database. *Nucleic Acids Res*, 32(Database issue):D138–D141.

[Bendtsen et al., 2004a] Bendtsen, J. D., Jensen, L. J., Blom, N., Heijne, G. V., and Brunak, S. (2004a). Feature-based prediction of non-classical and leaderless protein secretion. *Protein Eng Des Sel*, 17(4):349–356.

[Bendtsen et al., 2005] Bendtsen, J. D., Kiemer, L., Fausbøll, A., and Brunak, S. (2005). Non-classical protein secretion in bacteria. *BMC Microbiol*, 5:58.

[Bendtsen et al., 2004b] Bendtsen, J. D., Nielsen, H., von Heijne, G., and Brunak, S. (2004b). Improved prediction of signal peptides: SignalP 3.0. *J Mol Biol*, 340(4):783–795.

[Benjamini and Hochberg, 1995] Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *JOURNAL-ROYAL STATISTICAL SOCIETY SERIES B*, 57:289–289.

[Blobel, 2000] Blobel, G. (2000). Protein targeting (Nobel lecture). *Chembiochem.*, 1:86–102.

[Bolstad et al., 2003] Bolstad, B., Irizarry, R., Astrand, M., and Speed, T. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193.

[Bommarito et al., 2000] Bommarito, S., Peyret, N., and SantaLucia, J. (2000). Thermodynamic parameters for DNA sequences with dangling ends. *Nucleic Acids Res*, 28(9):1929–1934.

[Brockman et al., 2008] Brockman, W., Alvarez, P., Young, S., Garber, M., Giannoukos, G., Lee, W. L., Russ, C., Lander, E. S., Nusbaum, C., and Jaffe, D. B. (2008). Quality scores and snp detection in sequencing-by-synthesis systems. *Genome Res*, 18(5):763–770.

[Chen et al., 2004] Chen, G., Znosko, B. M., Jiao, X., and Turner, D. H. (2004). Factors affecting thermodynamic stabilities of RNA 3 x 3 internal loops. *Biochemistry*, 43(40):12865–12876.

[Clote et al., 2005] Clote, P., Ferré, F., Kranakis, E., and Krizanc, D. (2005). Structural RNA has lower folding energy than random RNA of the same dinucleotide frequency. *RNA*, 11(5):578–591.

[Cornette et al., 1987] Cornette, J. L., Cease, K. B., Margalit, H., Spouge, J. L., Berzofsky, J. A., and DeLisi, C. (1987). Hydrophobicity scales and computational techniques for detecting amphipathic structures in proteins. *J Mol Biol*, 195(3):659–685.

[Costa, 2007] Costa, F. F. (2007). Non-coding RNAs: lost in translation? *Gene*, 386(1-2):1–10.

[Creighton et al., 2009] Creighton, C. J., Reid, J. G., and Gunaratne, P. H. (2009). Expression profiling of micrornas by deep sequencing. *Brief Bioinform*, 10(5):490–497.

[Cronn et al., 2008] Cronn, R., Liston, A., Parks, M., Gernandt, D. S., Shen, R., and Mockler, T. (2008). Multiplex sequencing of plant chloroplast genomes using solexa sequencing-by-synthesis technology. *Nucleic Acids Res*, 36(19):e122.

[Crooks et al., 2004] Crooks, G. E., Hon, G., Chandonia, J.-M., and Brenner, S. E. (2004). WebLogo: a sequence logo generator. *Genome Res*, 14(6):1188–1190.

[Dayhoff and Schwartz, 1978] Dayhoff, M. O. and Schwartz, R. M. (1978). *Atlas of Protein Sequence and Structure*, volume 3 of *5 suppl.*, pages 353–358. Nat. Biomed. Res. Found., Washington D.C.

[Dempster et al., 1977] Dempster, A., Laird, N., Rubin, D., et al. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38.

[Dudoit et al., 2003] Dudoit, S., Shaffer, J., and Boldrick, J. (2003). Multiple Hypothesis Testing in Microarray Experiments. *STATISTICAL SCIENCE*, 18(1):71–103.

[Eddy, 2004] Eddy, S. R. (2004). Where did the BLOSUM62 alignment score matrix come from? *Nat Biotechnol*, 22(8):1035–1036.

[Eisen et al., 1998] Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868.

[Eisenberg et al., 1984] Eisenberg, D., Schwarz, E., Komaromy, M., and Wall, R. (1984). Analysis of membrane and surface protein sequences with the hydrophobic moment plot. *J Mol Biol*, 179(1):125–142.

[Emini et al., 1985] Emini, E. A., Hughes, J. V., Perlow, D. S., and Boger, J. (1985). Induction of hepatitis a virus-neutralizing antibody by a virus-specific synthetic peptide. *J Virol*, 55(3):836–839.

[Engelman et al., 1986] Engelman, D. M., Steitz, T. A., and Goldman, A. (1986). Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Annu Rev Biophys Biophys Chem*, 15:321–353.

[Falcon and Gentleman, 2007] Falcon, S. and Gentleman, R. (2007). Using GOstats to test gene lists for GO term association. *Bioinformatics*, 23(2):257.

[Felsenstein, 1981] Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–376.

[Feng and Doolittle, 1987] Feng, D. F. and Doolittle, R. F. (1987). Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol*, 25(4):351–360.

[Forsberg et al., 2001] Forsberg, R., Oleksiewicz, M. B., Petersen, A. M., Hein, J., Bøtner, A., and Storgaard, T. (2001). A molecular clock dates the common ancestor of European-type porcine reproductive and respiratory syndrome virus at more than 10 years before the emergence of disease. *Virology*, 289(2):174–179.

[Galperin and Koonin, 1998] Galperin, M. Y. and Koonin, E. V. (1998). Sources of systematic error in functional annotation of genomes: domain rearrangement, non-orthologous gene displacement and operon disruption. *In Silico Biol*, 1(1):55–67.

[Gill and von Hippel, 1989] Gill, S. C. and von Hippel, P. H. (1989). Calculation of protein extinction coefficients from amino acid sequence data. *Anal Biochem*, 182(2):319–326.

[Gonda et al., 1989] Gonda, D. K., Bachmair, A., Wünning, I., Tobias, J. W., Lane, W. S., and Varshavsky, A. (1989). Universality and structure of the N-end rule. *J Biol Chem*, 264(28):16700–16712.

[Guindon and Gascuel, 2003] Guindon, S. and Gascuel, O. (2003). A Simple, Fast, and Accurate Algorithm to Estimate Large Phylogenies by Maximum Likelihood. *Systematic Biology*, 52(5):696–704.

[Guo et al., 2006] Guo, L., Lobenhofer, E. K., Wang, C., Shippy, R., Harris, S. C., Zhang, L., Mei, N., Chen, T., Herman, D., Goodsaid, F. M., Hurban, P., Phillips, K. L., Xu, J., Deng, X., Sun, Y. A., Tong, W., Dragan, Y. P., and Shi, L. (2006). Rat toxicogenomic study reveals analytical consistency across microarray platforms. *Nat Biotechnol*, 24(9):1162–1169.

[Han et al., 1999] Han, K., Kim, D., and Kim, H. (1999). A vector-based method for drawing RNA secondary structure. *Bioinformatics*, 15(4):286–297.

[Hasegawa et al., 1985] Hasegawa, M., Kishino, H., and Yano, T. (1985). Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution*, 22(2):160–174.

[Hein, 2001] Hein, J. (2001). An algorithm for statistical alignment of sequences related by a binary tree. In *Pacific Symposium on Biocomputing*, page 179.

[Hein et al., 2000] Hein, J., Wiuf, C., Knudsen, B., Møller, M. B., and Wibling, G. (2000). Statistical alignment: computational properties, homology testing and goodness-of-fit. *J Mol Biol*, 302(1):265–279.

[Henikoff and Henikoff, 1992] Henikoff, S. and Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A*, 89(22):10915–10919.

[Hopp and Woods, 1983] Hopp, T. P. and Woods, K. R. (1983). A computer program for predicting protein antigenic determinants. *Mol Immunol*, 20(4):483–489.

[Ikai, 1980] Ikai, A. (1980). Thermostability and aliphatic index of globular proteins. *J Biochem (Tokyo)*, 88(6):1895–1898.

[Janin, 1979] Janin, J. (1979). Surface and inside volumes in globular proteins. *Nature*, 277(5696):491–492.

[Ji et al., 2008] Ji, H., Jiang, H., Ma, W., Johnson, D., Myers, R., and Wong, W. (2008). An integrated software system for analyzing ChIP-chip and ChIP-seq data. *Nature Biotechnology*, 26(11):1293–1300.

[Jukes and Cantor, 1969] Jukes, T. and Cantor, C. (1969). *Mammalian Protein Metabolism*, chapter Evolution of protein molecules, pages 21–32. New York: Academic Press.

[Kal et al., 1999] Kal, A. J., van Zonneveld, A. J., Benes, V., van den Berg, M., Koerkamp, M. G., Albermann, K., Strack, N., Ruijter, J. M., Richter, A., Dujon, B., Ansorge, W., and Tabak, H. F. (1999). Dynamics of gene expression revealed by comparison of serial analysis of gene expression transcript profiles from yeast grown on two different carbon sources. *Mol Biol Cell*, 10(6):1859–1872.

[Karplus and Schulz, 1985] Karplus, P. A. and Schulz, G. E. (1985). Prediction of chain flexibility in proteins. *Naturwissenschaften*, 72:212–213.

[Kaufman and Rousseeuw, 1990] Kaufman, L. and Rousseeuw, P. (1990). Finding groups in data. an introduction to cluster analysis. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics, New York: Wiley, 1990*.

[Kierzek et al., 1999] Kierzek, R., Burkard, M. E., and Turner, D. H. (1999). Thermodynamics of single mismatches in RNA duplexes. *Biochemistry*, 38(43):14214–14223.

[Kimura, 1980] Kimura, M. (1980). A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J Mol Evol*, 16(2):111–120.

[Klee and Ellis, 2005] Klee, E. W. and Ellis, L. B. M. (2005). Evaluating eukaryotic secreted protein prediction. *BMC Bioinformatics*, 6:256.

[Knudsen and Miyamoto, 2001] Knudsen, B. and Miyamoto, M. M. (2001). A likelihood ratio test for evolutionary rate shifts and functional divergence among proteins. *Proc Natl Acad Sci U S A*, 98(25):14512–14517.

[Kolaskar and Tongaonkar, 1990] Kolaskar, A. S. and Tongaonkar, P. C. (1990). A semi-empirical method for prediction of antigenic determinants on protein antigens. *FEBS Lett*, 276(1-2):172--174.

[Krogh et al., 2001] Krogh, A., Larsson, B., von Heijne, G., and Sonnhammer, E. L. (2001). Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol*, 305(3):567–580.

[Kyte and Doolittle, 1982] Kyte, J. and Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *J Mol Biol*, 157(1):105–132.

[Larget and Simon, 1999] Larget, B. and Simon, D. (1999). Markov chain monte carlo algorithms for the bayesian analysis of phylogenetic trees. *Mol Biol Evol*, 16:750–759.

[Leitner and Albert, 1999] Leitner, T. and Albert, J. (1999). The molecular clock of HIV-1 unveiled through analysis of a known transmission history. *Proc Natl Acad Sci U S A*, 96(19):10752–10757.

[Lloyd, 1982] Lloyd, S. (1982). Least squares quantization in PCM. *Information Theory, IEEE Transactions on*, 28(2):129–137.

[Longfellow et al., 1990] Longfellow, C. E., Kierzek, R., and Turner, D. H. (1990). Thermodynamic and spectroscopic study of bulge loops in oligoribonucleotides. *Biochemistry*, 29(1):278–285.

[Maeda et al., 2008] Maeda, N., Nishiyori, H., Nakamura, M., Kawazu, C., Murata, M., Sano, H., Hayashida, K., Fukuda, S., Tagami, M., Hasegawa, A., Murakami, K., Schroder, K., Irvine, K., Hume, D., Hayashizaki, Y., Carninci, P., and Suzuki, H. (2008). Development of a dna barcode tagging method for monitoring dynamic changes in gene expression by using an ultra high-throughput sequencer. *Biotechniques*, 45(1):95–97.

[Maizel and Lenk, 1981] Maizel, J. V. and Lenk, R. P. (1981). Enhanced graphic matrix analysis of nucleic acid and protein sequences. *Proc Natl Acad Sci U S A*, 78(12):7665–7669.

[Mathews et al., 2004] Mathews, D. H., Disney, M. D., Childs, J. L., Schroeder, S. J., Zuker, M., and Turner, D. H. (2004). Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of rna secondary structure. *Proc Natl Acad Sci U S A*, 101(19):7287–7292.

[Mathews et al., 1999] Mathews, D. H., Sabina, J., Zuker, M., and Turner, D. H. (1999). Expanded sequence dependence of thermodynamic parameters improves prediction of rna secondary structure. *J Mol Biol*, 288(5):911–940.

[Mathews and Turner, 2002] Mathews, D. H. and Turner, D. H. (2002). Experimentally derived nearest-neighbor parameters for the stability of RNA three- and four-way multibranch loops. *Biochemistry*, 41(3):869–880.

[Mathews and Turner, 2006] Mathews, D. H. and Turner, D. H. (2006). Prediction of RNA secondary structure by free energy minimization. *Curr Opin Struct Biol*, 16(3):270–278.

[McCaskill, 1990] McCaskill, J. S. (1990). The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–1119.

[McGinnis and Madden, 2004] McGinnis, S. and Madden, T. L. (2004). BLAST: at the core of a powerful and diverse set of sequence analysis tools. *Nucleic Acids Res*, 32(Web Server issue):W20–W25.

[Menne et al., 2000] Menne, K. M., Hermjakob, H., and Apweiler, R. (2000). A comparison of signal sequence prediction methods using a test set of signal peptides. *Bioinformatics*, 16(8):741–742.

[Meyer et al., 2007] Meyer, M., Stenzel, U., Myles, S., Prüfer, K., and Hofreiter, M. (2007). Targeted high-throughput sequencing of tagged nucleic acid samples. *Nucleic Acids Res*, 35(15):e97.

[Michener and Sokal, 1957] Michener, C. and Sokal, R. (1957). A quantitative approach to a problem in classification. *Evolution*, 11:130–162.

[Morin et al., 2008] Morin, R. D., O'Connor, M. D., Griffith, M., Kuchenbauer, F., Delaney, A., Prabhu, A.-L., Zhao, Y., McDonald, H., Zeng, T., Hirst, M., Eaves, C. J., and Marra, M. A. (2008). Application of massively parallel sequencing to microrna profiling and discovery in human embryonic stem cells. *Genome Res*, 18(4):610–621.

[Mortazavi et al., 2008] Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by rna-seq. *Nat Methods*, 5(7):621–628.

[Nielsen et al., 1997] Nielsen, H., Engelbrecht, J., Brunak, S., and von Heijne, G. (1997). Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Eng*, 10(1):1–6.

[Nielsen, 2007] Nielsen, K. L., editor (2007). *Serial Analysis of Gene Expression (SAGE): Methods and Protocols*, volume 387 of *Methods in Molecular Biology*. Humana Press.

[Nielsen et al., 2008] Nielsen, R., Pedersen, T., Hagenbeek, D., Moulos, P., Siersbæk, R., Megens, E., Denissov, S., Børgesen, M., Francoijs, K., Mandrup, S., et al. (2008). Genome-wide profiling of PPARγ: RXR and RNA polymerase II occupancy reveals temporal activation of distinct metabolic pathways and changes in RXR dimer composition during adipogenesis. *Genes & Development*, 22(21):2953.

[Parkhomchuk et al., 2009] Parkhomchuk, D., Borodina, T., Amstislavskiy, V., Banaru, M., Hallen, L., Krobitsch, S., Lehrach, H., and Soldatov, A. (2009). Transcriptome analysis by strand-specific sequencing of complementary dna. *Nucleic Acids Res*, 37(18):e123.

[Purvis, 1995] Purvis, A. (1995). A composite estimate of primate phylogeny. *Philos Trans R Soc Lond B Biol Sci*, 348(1326):405–421.

[Reinhardt and Hubbard, 1998] Reinhardt, A. and Hubbard, T. (1998). Using neural networks for prediction of the subcellular location of proteins. *Nucleic Acids Res*, 26(9):2230–2236.

[Rivas and Eddy, 2000] Rivas, E. and Eddy, S. R. (2000). Secondary structure alone is generally not statistically significant for the detection of noncoding RNAs. *Bioinformatics*, 16(7):583–605.

[Rose et al., 1985] Rose, G. D., Geselowitz, A. R., Lesser, G. J., Lee, R. H., and Zehfus, M. H. (1985). Hydrophobicity of amino acid residues in globular proteins. *Science*, 229(4716):834–838.

[Rost, 2001] Rost, B. (2001). Review: protein secondary structure prediction continues to rise. *J Struct Biol*, 134(2-3):204–218.

[Saitou and Nei, 1987] Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, 4(4):406–425.

[Sankoff et al., 1983] Sankoff, D., Kruskal, J., Mainville, S., and Cedergren, R. (1983). *Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparison*, chapter Fast algorithms to determine RNA secondary structures containing multiple loops, pages 93–120. Addison-Wesley, Reading, Ma.

[SantaLucia, 1998] SantaLucia, J. (1998). A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proc Natl Acad Sci U S A*, 95(4):1460–1465.

[Schechter and Berger, 1967] Schechter, I. and Berger, A. (1967). On the size of the active site in proteases. I. Papain. *Biochem Biophys Res Commun*, 27(2):157–162.

[Schechter and Berger, 1968] Schechter, I. and Berger, A. (1968). On the active site of proteases. 3. Mapping the active site of papain; specific peptide inhibitors of papain. *Biochem Biophys Res Commun*, 32(5):898–902.

[Schneider and Stephens, 1990] Schneider, T. D. and Stephens, R. M. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res*, 18(20):6097–6100.

[Schroeder et al., 1999] Schroeder, S. J., Burkard, M. E., and Turner, D. H. (1999). The energetics of small internal loops in RNA. *Biopolymers*, 52(4):157–167.

[Shapiro et al., 2007] Shapiro, B. A., Yingling, Y. G., Kasprzak, W., and Bindewald, E. (2007). Bridging the gap in RNA structure prediction. *Curr Opin Struct Biol*, 17(2):157–165.

[Siepel and Haussler, 2004] Siepel, A. and Haussler, D. (2004). Combining phylogenetic and hidden Markov models in biosequence analysis. *J Comput Biol*, 11(2-3):413–428.

[Smith and Waterman, 1981] Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. *J Mol Biol*, 147(1):195–197.

[Sneath and Sokal, 1973] Sneath, P. and Sokal, R. (1973). *Numerical Taxonomy*. Freeman, San Francisco.

[Stark et al., 2010] Stark, M. S., Tyagi, S., Nancarrow, D. J., Boyle, G. M., Cook, A. L., Whiteman, D. C., Parsons, P. G., Schmidt, C., Sturm, R. A., and Hayward, N. K. (2010). Characterization of the melanoma mirnaome by deep sequencing. *PLoS One*, 5(3):e9685.

[Sturges, 1926] Sturges, H. A. (1926). The choice of a class interval. *Journal of the American Statistical Association*, 21:65–66.

['t Hoen et al., 2008] 't Hoen, P. A. C., Ariyurek, Y., Thygesen, H. H., Vreugdenhil, E., Vossen, R. H. A. M., de Menezes, R. X., Boer, J. M., van Ommen, G.-J. B., and den Dunnen, J. T. (2008). Deep sequencing-based expression analysis shows major advances in robustness, resolution and inter-lab portability over five microarray platforms. *Nucleic Acids Res*, 36(21):e141.

[Tian et al., 2005] Tian, L., Greenberg, S., Kong, S., Altschuler, J., Kohane, I., and Park, P. (2005). Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences*, 102(38):13544–13549.

[Tobias et al., 1991] Tobias, J. W., Shrader, T. E., Rocap, G., and Varshavsky, A. (1991). The N-end rule in bacteria. *Science*, 254(5036):1374–1377.

[Tusher et al., 2001] Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci U S A*, 98(9):5116–5121.

[van Lunteren et al., 2008] van Lunteren, E., Spiegler, S., and Moyer, M. (2008). Contrast between cardiac left ventricle and diaphragm muscle in expression of genes involved in carbohydrate and lipid metabolism. *Respir Physiol Neurobiol*, 161(1):41–53.

[von Ahsen et al., 2001] von Ahsen, N., Wittwer, C. T., and Schütz, E. (2001). Oligonucleotide melting temperatures under PCR conditions: nearest-neighbor corrections for Mg(2+), deoxynucleotide triphosphate, and dimethyl sulfoxide concentrations with comparison to alternative empirical formulas. *Clin Chem*, 47(11):1956–1961.

[von Heijne, 1986] von Heijne, G. (1986). A new method for predicting signal sequence cleavage sites. *Nucl. Acids Res.*, 14:4683–4690.

[Welling et al., 1985] Welling, G. W., Weijer, W. J., van der Zee, R., and Welling-Wester, S. (1985). Prediction of sequential antigenic regions in proteins. *FEBS Lett*, 188(2):215–218.

[Wootton and Federhen, 1993] Wootton, J. C. and Federhen, S. (1993). Statistics of local complexity in amino acid sequences and sequence databases. *Computers in Chemistry*, 17:149–163.

[Workman and Krogh, 1999] Workman, C. and Krogh, A. (1999). No evidence that mRNAs have lower folding free energies than random sequences with the same dinucleotide distribution. *Nucleic Acids Res*, 27(24):4816–4822.

[Wyman et al., 2009] Wyman, S. K., Parkin, R. K., Mitchell, P. S., Fritz, B. R., O'Briant, K., Godwin, A. K., Urban, N., Drescher, C. W., Knudsen, B. S., and Tewari, M. (2009). Repertoire of micrornas in epithelial ovarian cancer as determined by next generation sequencing of small rna cdna libraries. *PLoS One*, 4(4):e5311.

[Yang, 1994a] Yang, Z. (1994a). Estimating the pattern of nucleotide substitution. *Journal of Molecular Evolution*, 39(1):105–111.

[Yang, 1994b] Yang, Z. (1994b). Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. *Journal of Molecular Evolution*, 39(3):306–314.

[Yang and Rannala, 1997] Yang, Z. and Rannala, B. (1997). Bayesian phylogenetic inference using DNA sequences: a Markov Chain Monte Carlo Method. *Mol Biol Evol*, 14(7):717–724.

[Zuker, 1989a] Zuker, M. (1989a). On finding all suboptimal foldings of an rna molecule. *Science*, 244(4900):48–52.

[Zuker, 1989b] Zuker, M. (1989b). The use of dynamic programming algorithms in rna secondary structure prediction. *Mathematical Methods for DNA Sequences*, pages 159–184.

[Zuker and Sankoff, 1984] Zuker, M. and Sankoff, D. (1984). Rna secondary structures and their prediction. *Bulletin of Mathemetical Biology*, 46:591–621.

[Zuker and Stiegler, 1981] Zuker, M. and Stiegler, P. (1981). Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res*, 9(1):133–148.

Part V

Index

Index